

Contraction-free proofs and finitary games for Linear Logic

André Hirschowitz¹

CNRS, Université de Nice - Sophia Antipolis, Nice, France

Michel Hirschowitz²

CEA - LIST, Saclay, France

Tom Hirschowitz^{3,4}

CNRS, Université de Savoie, Chambéry, France

Abstract

In the standard sequent presentations of Girard's Linear Logic [10] (LL), there are two "non-decreasing" rules, where the premises are not smaller than the conclusion, namely the cut and the contraction rules. It is a universal concern to eliminate the cut rule. We show that, using an admissible modification of the tensor rule, contractions can be eliminated, and that cuts can be simultaneously limited to a single initial occurrence.

This view leads to a consistent, but incomplete game model for LL with exponentials, which is *finitary*, in the sense that each play is finite. The game is based on a set of inference rules which does not enjoy cut elimination. Nevertheless, the cut rule is valid in the model.

Keywords: Linear logic, game semantics, contraction elimination.

1 Overview

In an effort to strengthen the connection between

- categories of sequent calculus proofs modulo cut elimination and
- categories of winning strategies,

¹ Email: ah@unice.fr

² Email: mh@lix.polytechnique.fr

³ Email: tom.hirschowitz@univ-savoie.fr

⁴ ANR projects Choco and MoDyFiable

we try to push forward the idea of *neutrality*, introduced by Delande and Miller [7] (see also our [13]). The idea there is to understand provability in MALL (Multiplicative Additive Linear Logic) as a graphical game⁵, as follows.

Positions of the game are graphs with edges labelled by formulae. The edges adjacent to each vertex form a sequent, the vertex itself representing a player trying to prove that sequent. Each move acts upon exactly one edge; its restrictions to each end of the edge correspond to inference rules in the sequent calculus. Finally, a formula A is valid when, on the following graph, the left-hand vertex (*Proponent* Δ) has a winning strategy against the right-hand vertex (*Opponent* \blacktriangle):

$$\Delta \xrightarrow{A} \blacktriangle .$$

In this paper, we extend this approach to exponential connectives, for which we recall the four standard rules:

$$\frac{\Gamma, A}{\Gamma, ?A} \quad \frac{\Gamma}{\Gamma, ?A} \quad \frac{\Gamma, ?A, ?A}{\Gamma, ?A} \quad \frac{? \Gamma, A}{? \Gamma, !A}$$

(where $? \Gamma$ denotes a list of $?$ formulae).

Interpreting these rules as moves in a graphical game raises the issue of infinite plays, for which deciding who wins is problematic.

In this paper, we explore the following way around this problem. We first replace the usual (right) tensor rule by an admissible variant

$$\frac{\Gamma, ?\Theta, A \quad \Delta, ?\Theta, B}{\Gamma, \Delta, ?\Theta, A \otimes B} \text{TENS}' ,$$

which is still somehow decreasing, yielding an equivalent set of inference rules which we call LL' . We then show that every sequent provable in LL' admits a *bounded* proof, i.e., a contraction-free one where cuts are reduced to at most one initial occurrence.

Using this, one may devise a graphical game for LL' without contraction, and derive from it a model of LL' provability, which is consistent (but incomplete). We sketch this, and explain why the result is not satisfactory: the model validates $A \multimap !A$.

We then investigate one possible explanation for this deficiency: the absence of n -ary dereliction. Restoring it leads to a set of inference rules which we call NLL , where $!A$ has the more standard decomposition $\&_{n \in \mathbb{N}} A^{\otimes n}$. The notion of a bounded proof makes sense in NLL , and we show that bounded proofs yield a model of LL' , albeit an incomplete one. We show that NLL does not satisfy cut elimination, but does satisfy admissibility of cut, i.e., if Γ, A and A^\perp, Δ have bounded proofs, then so does Γ, Δ .

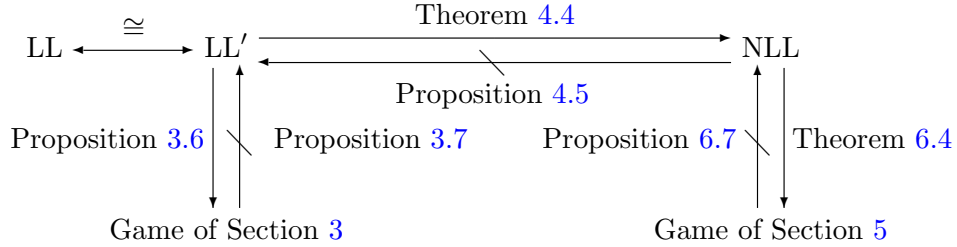
NLL yields a more satisfactory graphical game, in which winning strategies form a model of NLL , again consistent. Analogously to previous work [13], this model is not complete w.r.t. NLL (there is a winning strategy for $\perp \otimes \perp$ for instance).

⁵ Graphical game here means a game where positions are graphs. We do not give any formal definition.

However, we are confident that our notion of local strategy, defined in [13] for the MALL fragment, will extend smoothly to NLL, and yield a complete model.

We thus have (by composition) a graphical game model of LL (which, again, will not be complete for LL, because NLL itself is not). We finally prove that admissibility of cut holds in this model.

We may sum up our models of LL in a diagram



where an edge between two notions of validity indicates that the target is a model of the source, and a negated edge in the other direction indicates that the model is not complete. We conjecture that the vertical negated edges may be rectified by passing to local strategies in the sense of [13]. However, the horizontal negated arrow cannot be rectified.

Related work

The idea of a game semantics for logic goes back at least to Lorenzen [19], and to Blass [4] for linear logic. Blass' approach was extended by Abramsky et al. [1], Hyland and Ong [14], Abramsky and Melliès [2,20]. Laurent investigated the polarised case [18]. Among these standard approaches, only some handle exponential connectives, and among them none rule out infinite plays. Instead, they have to resort to smart criteria to decide who wins these infinite plays.

Delande and Miller's [7] game, our previous work [12,13], and Girard's *ludics* [11] of course were a source of inspiration for this paper.

Kashima [15] and Dyckhoff et al. [8,9] have already observed certain forms of contraction elimination in other settings.

Organisation of the paper

In Section 2, we prove that any sequent provable in LL has a bounded proof in LL'. In Section 3, we build a first graphical game from the rules of LL' minus contraction. We then prove that it yields a consistent model of LL' provability, which is however incomplete. In Section 4, we define our set of inference rules with n -ary dereliction, called NLL, and prove that it yields a model of LL' provability, which is again incomplete. We also show that NLL does not enjoy cut elimination. We then move on in Section 5 to define our graphical game for NLL, which we prove to yield a consistent model of NLL (hence NLL is itself consistent), which is again incomplete. In Section 7 we prove the cut rule to be admissible in the model.

Acknowledgments

We warmly thank Olivier Laurent for his constant “Linear Logic Hotline” and many useful examples and counterexamples. We also thank Paul-André Melliès for encouragements, and René David and Karim Nour for a very efficient consulting session. Finally, we thank the anonymous referees for very careful reading and helpful suggestions.

2 Bounded proofs in LL

In this section, we prove our result on linear logic, namely that any provable sequent admits a bounded proof.

First, we recall LL formulae, defined by the grammar

$$\begin{aligned} A, B, C, \dots ::= & \mathbf{0} \mid \mathbf{1} \mid A \otimes B \mid A \oplus B \mid ?A \\ & \mid \top \mid \perp \mid A \wp B \mid A \& B \mid !A, \end{aligned}$$

and decree that formulae on the first line are positive, the others being negative. De Morgan duality, or linear negation, A^\perp is defined as usual (sending a connective to that vertically opposed to it). Observe that we do not handle propositional variables; the sequent calculus part extends easily to variables, but for games, this is not yet clear to us. We use the usual priorities, e.g., $!A \multimap B \otimes C$ means $(!A) \multimap (B \otimes C)$. Finally, sequents Γ, Δ, \dots are lists of formulae, and we use $\Gamma \vdash \Delta$ as a notation for Γ^\perp, Δ , when visually easier.

Now, consider the following variant of the tensor rule

$$\frac{\Gamma, ?\Theta, A \quad \Delta, ?\Theta, B}{\Gamma, \Delta, ?\Theta, A \otimes B} \text{TENS}',$$

which goes back at least to Andreoli [3]. We observe that it is derivable in LL. Letting LL' denote the set of inference rules obtained by replacing the usual tensor rule with TENS' , the following should be clear:

Proposition 2.1 *Provability in LL' is equivalent to provability in LL.*

Thanks to rule TENS' , we further have:

Lemma 2.2 *For any formula A , the formula $\delta_A = !(A \multimap !A \otimes !A)$ is provable with neither cuts nor contractions in LL' , and furthermore for any Γ , the rule*

$$\frac{\Gamma, ?A^\perp, ?A^\perp}{\Gamma, ?A^\perp, \delta_A^\perp} \text{DUP}$$

is derivable without cuts nor contractions.

(Here, $A \multimap B$ denotes $A^\perp \wp B$, as usual.)

Proof. Here is a proof of δ_A :

$$\frac{\frac{\frac{?A^\perp, !A \quad ?A^\perp, !A}{?A^\perp, !A \otimes !A}}{!A \multimap (!A \otimes !A)}}{!(!A \multimap (!A \otimes !A))}.$$

Here is a derivation of DUP:

$$\frac{\frac{\frac{\Gamma, ?A, ?A}{\Gamma, ?A \wp ?A} \quad \frac{}{?A, !A^\perp}}{\Gamma, ?A, !A^\perp \otimes (?A \wp ?A)}}{\Gamma, ?A, !(A^\perp \otimes (?A \wp ?A))}.$$

□

We call *duplicators* the formulae of the shape δ_A . Using this, we construct *bounded* proofs for provable sequents in LL' , in the following sense.

Definition 2.3 A proof in LL' is *cc-free* when it is cut-free and contraction-free. It is *bounded* when it is either cc-free, or of the form

$$\frac{\frac{\pi_1}{A} \quad \frac{\pi_2}{A^\perp, \Gamma}}{\Gamma}$$

with π_1 and π_2 cc-free.

Lemma 2.4 For any sequent Γ provable in LL' , there is a list $\bar{\delta}$ of duplicators such that $\Gamma, \bar{\delta}^\perp$ admits a cc-free proof.

(Here $\bar{\delta}^\perp$ denotes the list of duals of formulae in $\bar{\delta}$.)

Proof. By cut elimination in LL , we may assume the given proof cut-free. The construction is then a mere induction on it, using the fact that duplicators are ! formulae. The non trivial case is contraction: from a given proof

$$\frac{\frac{\pi}{\Gamma, ?A, ?A}}{\Gamma, ?A}$$

one obtains by induction hypothesis a proof π' of $\Gamma, ?A, ?A, \bar{\delta}^\perp$, for some list of duplicators $\bar{\delta}$. One then derives (up to exchange):

$$\frac{\frac{\pi'}{\Gamma, ?A, ?A, \bar{\delta}^\perp}}{\Gamma, ?A, \bar{\delta}^\perp, \delta_A^\perp} \text{ DUP}$$

(This does not use contraction or cut.) For other cases, one just shows that a list of new hypotheses starting with ? do not hinder the derivation too much. This only adds weakenings and EXCHANGE rules, but no cuts nor contractions. \square

Finally, we conclude:

Theorem 2.5 *In LL' , each provable sequent admits a bounded proof.*

Proof. Consider a provable sequent Γ in LL' . By Lemma 2.4, Γ has a cc-free LL' proof π of $\Gamma, \bar{\delta}^\perp$, for some $\bar{\delta} = (\delta_1, \dots, \delta_n)$. So, the proof

$$\frac{\frac{\delta_1 \quad \dots \quad \delta_n}{\otimes (\bar{\delta})} \quad \frac{\frac{\pi}{\Gamma, \bar{\delta}^\perp}}{\mathfrak{A} (\bar{\delta})^\perp, \Gamma}}{\Gamma}$$

of Γ is bounded. \square

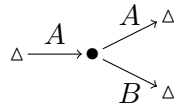
3 A naive game for LL'

We have shown that we can dispense with the contraction rule of LL , at the cost of a single initial cut and a modified tensor rule.

Starting from our previous game [13], we now build an extension of it to these rules.

3.1 Positions

As sketched above, the idea of positions in our game is as follows: positions are graphs, whose edges are labelled by formulae. We want to understand the neighborhood of a vertex, i.e., its adjacent edges, as its sequent. For instance, the vertex \bullet in the graph

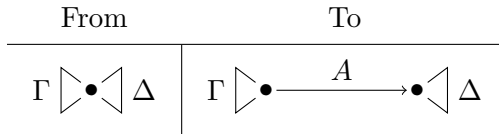


sees the sequent A, A^\perp, B , or equivalently $A \vdash A, B$.

The use of graphs may be understood by seeing the cut rule

$$\frac{\Gamma \vdash A \quad A \vdash \Delta}{\Gamma \vdash \Delta}$$

as a move



between graphs. Starting from the single vertex graph, after a finite number of cuts, one may reach any connected and acyclic graph (in the undirected sense). Hence our definition:

Definition 3.1 A *position* in our game is a *punctured, directed, signed, normally labelled tree*.

By *punctured*, we mean that a vertex is selected; we say that this vertex *holds the token*. By *directed* we mean that edges are ordered pairs of vertices. By *signed*, we mean that a function from vertices to $\{O, P\}$ is selected. Vertices labelled O , called *opponents*, are pictured by \blacktriangle , others, called *proponents*, are pictured by \triangle . By *labelled*, we mean that a function from edges to formulae is selected, and by *normally labelled*, we mean that the selected formulae are positive. Finally by *tree*, we mean that the underlying undirected graph is a tree.

We will also consider non-normally labelled trees, as a notation for the position obtained by normalisation: for each edge labelled with a negative formula, reverse altogether the edge and the formula.

There is an obvious notion of isomorphism, which turns positions into a groupoid. Unfortunately positions may have symmetries (automorphisms). In the following, we consider positions up to isomorphism.

For each vertex, its adjacent edges determine (up to reordering) its *sequent*.

3.2 Moves

We now define our moves in Figure 1. Moves are the edges of a (huge) graph with positions as vertices, or equivalently a reduction relation on positions.

Moves go left to right (in each row), and the vertex v holding the token, pictured as framed, *plays* the move, or is *active*. The other vertex v' is *passive*.

The move is defined regardless of v and v' being proponents or an opponents. Thus, to avoid notational clutter, we define moves as a relation between graphs, the polarity $\{O, P\}$ being inherited from the initial position. In the figure, \bullet thus represents vertices regardless of their polarity.

On the left, the *broken* edge or formula is shown. It determines two subtrees, which are pictured as triangles, and named to emphasise the analogy with inference rules. In the initial position of the tensor move (third line), there are several triangles connected to the left-hand vertex v : they denote subtrees which have v as only vertex in common.

On the first row, we have *negative* moves, where A is required to be positive. The active vertex v passes the token along an input edge. In some sense, v has a negative formula, and asks the vertex at the other end of this formula to break it. Thanks to acyclicity of our positions, the token will come back, if ever, through the same edge, breaking it at the same time. In the case of the negative formula \top , the token will never come back, since there is no right introduction rule for $\mathbf{0}$. In this sense, the corresponding negative move is winning.

Then we have positive moves. The active vertex may act on each of its output formulae, except the $\mathbf{0}$'s, and conditionally for the $\mathbf{1}$'s. The action is simple on formulae of the form $A \oplus B$: it just changes the formula either into A or into B .

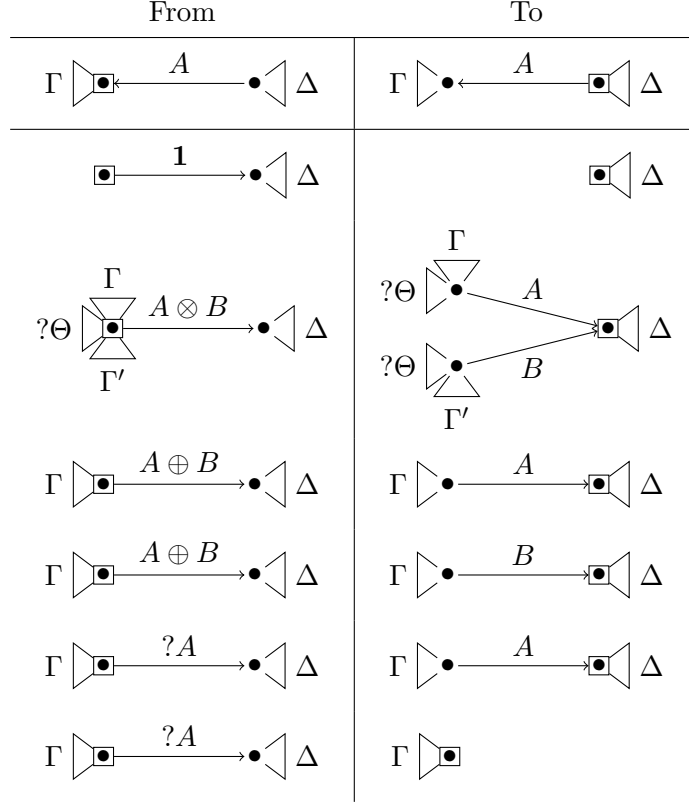


Fig. 1. Naive moves

On $?A$, the vertex may change it to A , or brutally erase $?A$, together with its whole subtree. The action is pretty simple on the formula $\mathbf{1}$: if (and only if) that edge is its sole adjacent edge, the vertex may pass the token along that formula, and delete itself and the edge from the tree.

It remains to explain the positive move in the case of a tensor formula. This is the most complex move. Here our active vertex, say v , has an output edge e labelled with $A \otimes B$, ending at the neighbour vertex v' . When deleting e , we get a tree Δ on the v' side, and another tree on the v side, which v has to split into three subtrees Γ, Γ' , and $? \Theta$, sharing only their root v . Furthermore, the subtree $? \Theta$ must exclusively consist of outputs of the form $?C$. Then our vertex v becomes two vertices v_1 and v_2 , where v_1 inherits A (linked to v'), Γ and a copy of $? \Theta$, while v_2 inherits B (also linked to v'), Γ' and a second copy of $? \Theta$. On the other side, v' keeps Δ unchanged, and inherits both input formulae A and B , plus the token.

Our formalism allows to define cut moves and contraction moves (e.g., by changing $?A$ to $?A \wp ?A$), but we leave them out of the picture.

3.3 Plays, strategies, validity

We now define plays and (winning) strategies, and show that the latter yield a consistent model of LL' provability (which is incomplete).

Having defined moves, plays on a position U follow: they are (directed, possibly countable) paths from U in the graph of positions and moves, or equivalently

reduction sequences. They happen to be finite:

Lemma 3.2 *In the above graph of positions and moves, there is no infinite path (i.e., in our game, all plays are finite).*

Proof. This is a consequence of our later Theorem 5.1. \square

For a position U , we then define a *strategy* to be a *welcoming*, prefix-closed set of plays from U , containing the empty play. By *welcoming*, we mean that the strategy accepts any move from opponents. In a strategy S , a play is *maximal* when S contains no extension of it.

Definition 3.3 A strategy is *winning* if, in all its maximal positions, the token is held by an opponent.

We then define validity. This is a bit more delicate, because the game does not feature contraction, so for proofs to yield winning strategies, we account for the needed initial cut:

Definition 3.4 A formula A is *GLL'-valid* (G stands for ‘‘Game’’) iff there exists a tree U such that for all trees V , the vertices of U have a winning strategy against those of V in the graph

$$U \triangleright_{\Delta} \xrightarrow{A} \blacktriangleleft V$$

wherever the token is placed initially.

Using the same methods as in Section 6 below, one may show that GLL'-validity yields a consistent model of LL':

Theorem 3.5 *GLL'-validity is consistent, i.e., two dual formulae are not both GLL'-valid.*

Proposition 3.6 *Any formula provable in LL' is GLL'-valid.*

The proofs are as in Section 6, so we do not repeat them here.

We consider this game unsatisfactory, because its exponentials are terribly non standard:

Proposition 3.7 *The formula $A \multimap !A$ is GLL'-valid.*

We first observe:

Lemma 3.8 *Axioms $A \vdash A$ are GLL'-valid.*

Then:

Proof of Proposition 3.7 One reaches a sequent $A \vdash !A$, where, roughly, one may pass the token along $!A$, and then either be erased (and hence win), or become $A \vdash A$ (and win again). \square

The remainder of the paper patches this deficiency by allowing n -ary dereliction. We first do this for inference rules.

4 NLL

Let us start from the rules of LL' . We want to patch them to add n -ary dereliction. This means that instead of contraction, weakening, and dereliction, we use the rules:

$$\frac{\Gamma, \perp}{\Gamma, ?A} \quad \frac{\Gamma, A}{\Gamma, ?A} \quad \cdots \quad \frac{\Gamma, \wp^n A}{\Gamma, ?A} \quad \cdots$$

with $\wp^0 A = \perp$ and $\wp^{n+1} A = \wp^n A \wp A$.

To preserve the symmetry between dual connectives, we use for $!$ the rule

$$\frac{\Delta, \mathbf{1} \quad \Delta, A \quad \dots \quad \Delta, \otimes^n A \quad \dots}{\Delta, !A}$$

which has infinitely many premises.

The idea behind these rules is to encode $!A$ as an infinitary additive conjunction $\&_{n \in \mathbb{N}} A^{\otimes n}$, which is an old idea, at least as old as Lafont's work on categorical models of LL [16,17].

This yields a set of inference rules, which we sum up in Figure 2 and call NLL. Since we have infinitary rules, let us precisely define proofs.

Definition 4.1 A proof in NLL is a tree whose branches are all finite, and whose nodes are labelled with inference rules as usual.

However, we are mainly interested in *bounded* proofs, in the following sense:

Definition 4.2 A proof in NLL is *bounded* when it is either cut-free, or of the form

$$\frac{\frac{\pi_1}{A} \quad \frac{\pi_2}{A^\perp, \Gamma}}{\Gamma} \tag{1}$$

with π_1 and π_2 cut-free.

We now show that:

- NLL is a model of LL provability;
- This model is incomplete, i.e., the converse does not hold;
- NLL does not enjoy cut elimination.

Lemma 4.3 *Axioms $A \vdash A$ have bounded proofs in NLL.*

Theorem 4.4 *Each provable sequent in LL' admits a bounded proof in NLL.*

Proof. By induction on a given bounded proof in LL' , we construct a bounded proof (of the same sequent) in NLL. All cases but promotion are obvious by induction hypothesis (because the rules of LL' are almost the same as those of NLL). For promotion, assume given a proof of $? \Gamma, A$. We need to find an NLL proof of each $? \Gamma, \otimes^n A$: for each n , we split the n -fold tensor, and reduce to proving $? \Gamma, A$ or $? \Gamma, \mathbf{1}$, which are both have bounded proofs in NLL (the former by induction hypothesis,

Positive rules	Negative rules
$\overline{\mathbf{1}}$	$\frac{\Gamma}{\Gamma, \perp}$
	$\overline{\Gamma, \top}$
$\frac{\Gamma, ?\Theta, A \quad \Delta, ?\Theta, B}{\Gamma, \Delta, ?\Theta, A \otimes B}$	$\frac{\Gamma, A, B}{\Gamma, A \wp B}$
$\frac{\Gamma, A}{\Gamma, A \oplus B} \quad \frac{\Gamma, B}{\Gamma, A \oplus B}$	$\frac{\Gamma, A \quad \Gamma, B}{\Gamma, A \& B}$
$\frac{\Gamma, \wp^n A}{\Gamma, ?A}$	$\frac{\Gamma, \mathbf{1} \quad \dots \quad \Gamma, \otimes^n A \quad \dots}{\Gamma, !A}$
$\frac{\Gamma, B, A, \Delta}{\Gamma, A, B, \Delta}$	$\frac{\Gamma, A \quad \Delta, A^\perp}{\Gamma, \Delta}$

Fig. 2. Inference rules for NLL

the latter straightforwardly). Since we only introduce tensor and $\mathbf{1}$ rules, this results in a bounded proof. \square

We have thus shown that bounded proofs in NLL are a model of LL' provability. However, the converse does not hold, thanks to an example of Olivier Laurent:

Proposition 4.5 *LL' is not a model of NLL.*

Proof. The sequent $\perp \oplus (X^\perp \wp ?X^\perp), !X$ has a bounded proof in NLL, as follows:

$$\frac{\frac{\dots}{\perp, \mathbf{1}} \quad \frac{\frac{X^\perp, X}{X^\perp \wp ?X^\perp, X}}{\perp \oplus (X^\perp \wp ?X^\perp), X} \quad \dots}{\perp \oplus (X^\perp \wp ?X^\perp), !X}$$

However, it is not provable in LL. There could be a problem since it uses a variable, but taking $X = \perp \otimes \perp$ makes everything work the same. \square

Finally:

Proposition 4.6 *NLL does not enjoy cut elimination.*

Proof. Consider the sequent $X \& ((X \& Y) \wp Y) \vdash ?(X \& Y)$. It admits the following LL' proof:

$$\frac{\frac{\frac{X \vdash X}{X \vdash ?(X \& Y), X}}{X \& ((X \& Y) \wp Y) \vdash ?(X \& Y), X} \quad \frac{\frac{Y \vdash Y \quad \frac{X \& Y \vdash X \& Y}{X \& Y \vdash ?(X \& Y)}}{((X \& Y) \wp Y) \vdash ?(X \& Y), Y}}{X \& ((X \& Y) \wp Y) \vdash ?(X \& Y), Y}}{X \& ((X \& Y) \wp Y) \vdash ?(X \& Y), (X \& Y)}}{X \& ((X \& Y) \wp Y) \vdash ?(X \& Y), ?(X \& Y)}}{X \& ((X \& Y) \wp Y) \vdash ?(X \& Y),}$$

hence has a bounded proof in NLL by Theorem 4.4. However, careful inspection shows that it has no cut-free proof. Again, we do this using variables, which are not in the syntax. It all works the same by taking X and Y such that none entails the other, e.g., $X = \perp \otimes \top$ and $Y = 1 \wp \mathbf{0}$. \square

Nevertheless, we have:

Proposition 4.7 *NLL features admissibility of cut, i.e., if Γ, A and A^\perp, Δ both have bounded proofs, then so does Γ, Δ .*

Proof. Assuming first that the given bounded proofs have the shape:

$$\frac{\frac{\pi_1}{\Gamma, A, B} \quad \frac{\pi_2}{B^\perp}}{\Gamma, A} \quad \frac{\frac{\pi_3}{\Delta, A^\perp, C} \quad \frac{\pi_4}{C^\perp}}{\Delta, A^\perp}$$

with the π_i 's cut-free. We then form the bounded proof:

$$\frac{\frac{\frac{\frac{\pi_1}{\Gamma, A, B} \quad \frac{\pi_3}{\Delta, A^\perp, C}}{\Gamma, \Delta, (A \otimes A^\perp), B, C} \quad \frac{\frac{\pi_2}{B^\perp} \quad \frac{\pi_4}{C^\perp}}{B^\perp \otimes C^\perp} \quad A^\perp, A}}{\Gamma, \Delta, (A \otimes A^\perp) \wp B \wp C} \quad \frac{A^\perp, A}{(A^\perp \wp A) \otimes B^\perp \otimes C^\perp}}{\Gamma, \Delta.}$$

Cases where not both proofs start with a cut behave essentially the same. \square

We now proceed to define a graphical game interpretation of NLL.

5 A graphical game for NLL

The positions of our game are exactly as in Section 3. Moves are as in Figure 1, with both exponential moves replaced by the single move scheme:

$$\begin{array}{c} \text{From} \qquad \qquad \qquad \text{To} \\ \hline \Gamma \triangleright \bullet \xrightarrow{?A^\perp} \bullet \triangleleft \Delta \quad \Bigg| \quad \Gamma \triangleright \bullet \xleftarrow{\otimes^n A} \bullet \triangleleft \Delta \end{array}$$

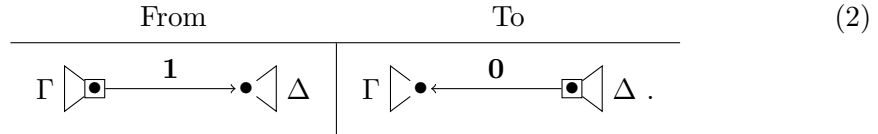
(using the conventions of Figure 1).

Plays are as in Section 3: (directed, possibly countable) paths in the graph of positions and moves. As in Section 3, we have: finite:

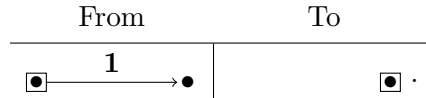
Theorem 5.1 *All plays are finite.*

We prove this by adapting to our graphical setting the syntactic techniques of David and Nour [5,6].

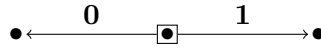
We will need a new move, which we call the *exotic* move:



Call a position U *strongly normalizable* (sn for short) if, even if we add the above exotic move, all plays from U are finite. Why do we need a new move? We want to be sure that if a position is sn, so are all its subpositions, in the sense of subgraphs. And the problem with this is that, although most moves on a subposition come from a move on the full position, this is not the case for a $\mathbf{1}$ -move. Indeed, consider the bare $\mathbf{1}$ -move



If we embed the initial position into the bigger



we obtain a position without any possible move. Our exotic move allows the token to be passed to the right. The intuition is that it transforms $\mathbf{1}$ -moves into “one-way” moves, merely passing the token along $\mathbf{1}$ without any hope for return. Which has roughly the same effect as deleting the edge.

Thanks to the exotic move, we have:

Lemma 5.2 *Any subposition of an sn position is again sn.*

Now, consider a tuple $U = (U, v, a_1, \dots, a_n)$ where U is a position with marked vertices a_1, \dots, a_n , and v is the vertex owning the token. Consider also a tuple $V = ((V_1, b_1), \dots, (V_n, b_n))$ of positions (b_i is the node having the token), and a formula A . From these data, we build a new position $W = W(U, V, A)$ by taking the union of U with the V_i 's, and adding n new edges from a_i to b_i for each i , labelling these with A (which could be negative), and putting the token at v .

Using this notation, the theorem follows (by an easy induction on the number of vertices) from the following lemma:

Lemma 5.3 *If U and the V_i 's are sn, then so is W .*

Proof. We proceed by lexicographic induction on the triple (A, U, n) , using the following orderings:

Formulae: We use the ordering determined by the following four rules:

- A formula is greater than each of its subformulae and their duals;
- $\mathbf{1}$ is greater than \top ;
- A formula $?C$ is greater than $\mathfrak{A}^n C$ for each n (and similarly for $!$ and \otimes);
- A negative formula is greater than its linear negation.

Positions: We order sn positions by putting $U \geq V$ iff there is a path in the graph of moves, or equivalently a reduction sequence, from U to V . Since we limit this to sn positions, it is well-founded.

Numbers: Finally, we use the standard ordering for natural numbers.

In order to prove that there is no infinite play starting at W , it is enough to show that all positions reached from W after one move are sn. We proceed by a case analysis on this first move.

If it is a move inside U : $U \rightarrow U'$, we apply the induction hypothesis to the new position W_0 . We have to explain how it has the required form. Indeed, we have $W_0 = W(U', V', A)$, where:

- the marked points in U' remain the same, except in the case of a $\mathbf{1}$ move, where some a_i 's may be deleted, and in the case of a \otimes move where some a_i 's may be duplicated;
- V' is like V , except in case the move is a $\mathbf{1}$ move, where possibly some V_i 's have to be deleted, and in case of a \otimes move, where some V_i 's have to be duplicated.

Hence in the present case, n may increase, but A is constant, and U strictly decreases.

Suppose now $v = a_j$ and our move $W \rightarrow W_0$ is on the j -th A edge between a_j and b_j .

We treat separately the most delicate case where $A = B \otimes C$. We prove that W_0 is sn by applying three times the induction hypothesis as follows. Recalling names from the description of the tensor move, let U' be the part of U keeping Γ , and U'' that keeping Δ . Now, leaving out j , each a_i uniquely goes to either U' or U'' after our move. We let V' be the tuple of all V_i 's with a_i in U' , and V'' be the tuple of all V_i 's with a_i in U'' . Then, by induction hypothesis, $W' = W(U', V', A)$ is sn because A remains the same, U' is smaller or equal to U , and n has strictly decreased. Similarly, $W'' = W(U'', V'', A)$ is sn. Then consider $W_1 = W(W', V_j, B)$ is also sn by induction hypothesis because $B < B \otimes C$. Finally, $W_0 = W(W'', W_1, C)$ is sn by induction hypothesis for the same reason.

For all other cases, we apply the inductive hypothesis twice: one for the new V , with smaller n , and one for the new U , with a smaller A . □

By the way, Lemma 3.2 follows by remarking that any play in the game of Section 3 is simulated by a play in the new game (with the exotic move (2)).

6 Soundness

So we have a game for NLL. Let us now show that it defines a model of NLL (and hence of LL). The definition of (winning) strategies is exactly as in Section 3,

thanks to Theorem 5.1. As in Section 3, we put:

Definition 6.1 A formula A is *GNLL-valid* (again, G stands for “Game”) iff there exists a tree U such that for all trees V , the vertices of U have a winning strategy against those of V in the graph

$$U \triangleright_{\Delta} \xrightarrow{A} \blacktriangleleft V$$

wherever the token is placed initially.

We immediately have consistency:

Theorem 6.2 *A formula and its dual are not both GNLL-valid.*

Proof. Given two trees U and V as above, and strategies S and S' for them, they both contain the empty play so $S \cap S'$ is non-empty. By Theorem 5.1, we have a maximal play in $S \cap S'$. In its final position, the token is held by some vertex v . If v is a proponent then S is not winning, otherwise S' is not winning. \square

As a consequence, since NLL has a consistent model, we have:

Corollary 6.3 *NLL is consistent.*

We now prove:

Theorem 6.4 *GNLL-validity is a model of NLL (bounded) provability.*

We start with:

Lemma 6.5 *A strategy is winning iff it is total, i.e., for each reached position where a proponent holds the token, it has an extension by a move.*

We then observe:

Lemma 6.6 *For any position U , equipped with, for each proponent vertex $v \in U$, a cut-free NLL proof π_v of its sequent, there exists a winning strategy on U .*

Proof. Before constructing such a strategy, let us observe that the following rules are admissible in NLL, and moreover for any cut-free proof of their premise, there is a cut-free proof of their conclusion:

$$\frac{\Gamma, \perp}{\Gamma} \quad \frac{\Gamma, A \wp B}{\Gamma, A, B} \quad \frac{\Gamma, A \& B}{\Gamma, A} \quad \frac{\Gamma, A \& B}{\Gamma, B} \quad \frac{\Gamma, !A}{\Gamma, \otimes^n A}.$$

We say that going from premise to conclusion in one of these rules is an *anodyne* modification.

Assume now given a position U , and a cut-free proof π_v for each proponent $v \in U$, and let us construct a winning strategy on U . We describe this by defining a new graph \mathcal{G} , which embeds into the graph of positions and moves. It has as vertices all positions U equipped with a cut-free proof π_v for each proponent $v \in U$. We now define edges from any such U .

First assume the token is held by opponents in U . After each move $U \rightarrow V$, if the token is still held by opponents in V , then proponent sequents remain unchanged,

and we add an edge $U \rightarrow V$ in \mathcal{G} . On the other hand, if the token is now held by a proponent, the corresponding sequent may have been anodynely modified. As observed above, we still may choose a cut-free proof for this sequent, as well as for the unmodified ones, and we add an edge $U \rightarrow V$ in \mathcal{G} .

Now if the token is held by proponents in U , we consider the first step of the proof of the involved sequent. If it is a positive rule, we consider the corresponding move $U \rightarrow V$. In V , we still have cut-free proofs for all proponents' sequents, and we add an edge $U \rightarrow V$ in \mathcal{G} . If the proof starts with a negative rule, we add as an edge in \mathcal{G} the move passing the token along the corresponding formula (in the new position, proponents' sequents are unchanged, hence we still have cut-free proofs for them).

But this graph \mathcal{G} determines a strategy on each of its vertices, since it accepts all opponent moves. And furthermore this strategy is total thanks to proofs, hence it is winning by Lemma 6.5. \square

Proof of Theorem 6.4 Consider any formula B with a bounded proof in NLL. We have to choose a graph U such that for all V , vertices in the position

$$U \triangleright_{\Delta} \xrightarrow{B} \blacktriangleleft V \quad (3)$$

have a winning strategy.

If the given proof of B is cut-free, then Lemma 6.6 provides a winning strategy for $U = \Delta$, the single vertex (for any V and initial placement of the token).

Otherwise the bounded proof has the shape

$$\frac{\frac{\pi_1}{A} \quad \frac{\pi_2}{A^{\perp}, B}}{B.}$$

We then choose U to be

$$\Delta \xrightarrow{A} \Delta$$

so that for any V , the position (3) becomes

$$\Delta \xrightarrow{A} \Delta \xrightarrow{B} \blacktriangleleft V.$$

We then assign them the cut-free proofs π_1 and π_2 , and obtain by Lemma 6.6 a winning strategy for them (for any placement of the token). \square

However, we have:

Proposition 6.7 *GNLL-validity is complete neither w.r.t. LL, nor w.r.t. NLL.*

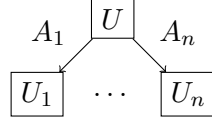
Proof. Indeed, exactly as in the game for MALL [13], $\perp \otimes \perp$ is GNLL-valid but provable neither in LL nor in NLL. \square

We expect that *local* strategies [13] will remedy incompleteness.

7 Admissibility of cut

We now define a notion of GNLL-validity for arbitrary sequents, which extends that for formulae, and prove that the cut rule holds in our model, i.e., if two sequents Γ, A and A^\perp, Δ are GNLL-valid, then so is Γ, Δ .

Definition 7.1 A sequent $\Gamma = (A_1, \dots, A_n)$ is *GNLL-valid* iff there is a position U such that for all tuples of positions U_1, \dots, U_n , the vertices in U have a winning strategy S on the position



The pair (U, S) is then a witness for the GNLL-validity of Γ . We say that U is its *base position*.

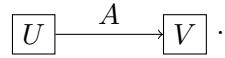
Theorem 7.2 *The cut rule holds, i.e., if two sequents Γ, A and A^\perp, Δ are GNLL-valid, then so is Γ, Δ .*

We first consider slightly generalised positions U , with three teams instead of two, say, P , P' , and O . The teams P and P' are to be thought of as a partition of proponents. For such a generalised position U , call (U, P) the non-generalised position with P as proponents and $P' \cup O$ as opponents; and similarly for (U, P') , $(U, P \cup P')$, and (U, O) . We have:

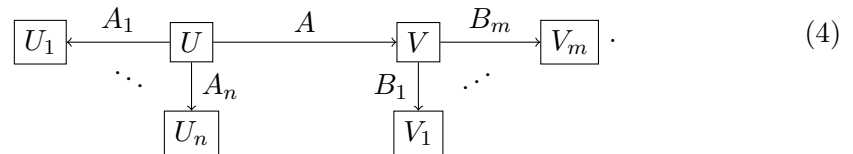
Lemma 7.3 *For a generalised position U , given winning strategies S on (U, P) and S' on (U, P') , the set of plays $S \cap S'$ is a winning strategy on $(U, P \cup P')$.*

Proof. It is a strategy because it is obviously prefix-closed, accepts all moves by O . Furthermore, it is total, hence winning by the previous lemma. \square

Proof of Theorem 7.2 Assume given two GNLL-valid sequents Γ, A and A^\perp, Δ , with associated base positions U and V , and with $\Gamma = (A_1, \dots, A_n)$ and $\Delta = (B_1, \dots, B_m)$. Then, choose as a base for Γ, Δ the position



Then, for any tuple $(U_1, \dots, U_n, V_1, \dots, V_m)$, consider the corresponding position



The GNLL-validity of Γ, A induces a winning strategy S for U , and that of A^\perp, Δ induces a winning strategy S' for V . This yields the winning strategy $S \cap S'$ for $U \cup V$ on (4). Hence Γ, Δ is GNLL-valid. \square

References

- [1] Abramsky, S. and R. Jagadeesan, *Games and full completeness for multiplicative linear logic*, Journal of Symbolic Logic **59** (1994), pp. 543–574.
- [2] Abramsky, S. and P.-A. Melliès, *Concurrent games and full completeness*, in: *Proceedings of the fourteenth annual symposium on Logic In Computer Science* (1999), pp. 431–442.
- [3] Andreoli, J.-M., “Proposition pour une Synthèse des Paradigmes de la Programmation Logique et de la Programmation par Objets,” Ph.D. thesis, Université Paris 6 (1990).
- [4] Blass, A., *A game semantics for linear logic*, Annals of Pure and Applied Logic **56** (1992), pp. 183–220.
- [5] David, R., *Normalization without reducibility*, Annals of Pure and Applied Logic **107** (2001).
- [6] David, R. and K. Nour, *A short proof of the strong normalization of classical natural deduction with disjunction*, Journal of Symbolic Logic **68** (2003).
- [7] Delande, O. and D. Miller, *A neutral approach to proof and refutation in MALL*, in: *LICS '08*, 2008, pp. 498–508.
- [8] Dyckhoff, R., *Contraction-free sequent calculi for intuitionistic logic*, The Journal of Symbolic Logic **57** (1992), pp. 795–807.
URL <http://www.jstor.org/stable/2275431>
- [9] Dyckhoff, R. and S. Negri, *Admissibility of structural rules for extensions of contraction-free sequent calculi*, Logic Journal of the IGPL **9** (2001).
- [10] Girard, J.-Y., *Linear logic*, Theoretical Computer Science **50** (1987), pp. 1–102.
- [11] Girard, J.-Y., *Locus solum: From the rules of logic to the logic of rules*, Mathematical Structures in Computer Science **11** (2001), pp. 301–506.
- [12] Hirschowitz, A., M. Hirschowitz and T. Hirschowitz, *Playing for truth on mutable graphs*, Talk at the Colloquium in honor of Gérard Huet’s 60th birthday, ENS, Paris (2007), www.lama.univ-savoie.fr/~hirschowitz/papers/H3.pdf.
- [13] Hirschowitz, A., M. Hirschowitz and T. Hirschowitz, *Towards topological games (long version)* (2008), www.lama.univ-savoie.fr/~hirschowitz/papers/mall-long.pdf.
- [14] Hyland, M. and L. Ong, *Fair games and full completeness for multiplicative linear logic without the mix-rule* (1993), preprint.
- [15] Kashima, R., *Contraction-elimination for implicative logics*, Annals of Pure and Applied Logic **84** (1997), pp. 17–39.
- [16] Lafont, Y., *The linear abstract machine*, Theor. Comput. Sci. **59** (1988), pp. 157–180.
- [17] Lafont, Y., *The linear abstract machine (corrigenda)*, Theor. Comput. Sci. **62** (1988), pp. 327–328.
- [18] Laurent, O., “Étude de la polarisation en logique,” Thèse de doctorat, Université Aix-Marseille II (2002).
- [19] Lorenzen, P., *Ein dialogisches konstruktivitätskriterium*, in: *Infinitistic Methods: Proceed. Symp. Foundations of Math.* (1961), pp. 193–200.
- [20] Melliès, P.-A., *Asynchronous games 4: A fully complete model of propositional linear logic*, in: P. Panangaden, editor, *Proceedings of the Twentieth Annual IEEE Symp. on Logic in Computer Science, LICS 2005* (2005), pp. 386–395.