Lecture Notes on Validity

15-836: Substructural Logics
Frank Pfenning

Lecture 9
September 26, 2023

1 Introduction

So far we have drawn strict boundaries between ordered, linear, and structural logic. To make linear logic more expressive we have used recursion because it is quite natural from the programming perspective. In Lecture 2 we briefly mentioned and showed rules for the exponential modality $!A$ which is subject to weakening and contraction.

In this lecture we explain that $!A$ is the result of a general construction that can be carried out for other logics as well. In structural logics, for example, it is usually written as $\Box A$, expressing that A is necessarily true or A is true in all possible worlds. The proof theory for $!A$ when viewed from this perspective is somewhat more pleasant, but ultimately still not entirely satisfactory. We will come back to this in the next lecture to improve and in the process generalize it even further.

2 Girard's Exponential

Girard's [1987] exponential modality, in the intuitionistic setting [Girard and Lafont, 1987], can be defined in the sequent calculus with the following rules.

$$\frac{!\Delta \vdash A}{!\Delta \vdash !A}\ !R \qquad \frac{\Delta, A \vdash C}{\Delta, !A \vdash C}\ !L$$

$$\frac{\Delta, !A, !A \vdash C}{\Delta, !A \vdash C}\ \textsf{contract} \qquad \frac{\Delta \vdash C}{\Delta, !A \vdash C}\ \textsf{weaken}$$

Here, $!\Delta$ means that every antecedent in Δ has the form $!B$. With these rules we can obtain as many copies of A from $!A$ as we want.

3 Andreoli's Exponential

Andreoli [1992] introduced what he calls a *dyadic* system for linear logic where we have two distinct forms of antecedents (later also given in its intuitionistic version [Barber, 1996]). From our perspective, one collection of antecedents is *structural* and the other is *linear*. We write such a sequent as

$$\Gamma \,;\, \Delta \vdash A$$

Here, Γ represents a set and Δ a multiset. For Andreoli, this was mostly a technical device; we will justify it as the result of studying *validity*.

What does it mean for a proposition A to be *valid* as opposed to merely *true*? A slogan from an earlier lecture may be helpful:

> Truth is ephemeral, validity forever.

A proposition such as "it is raining" may be true in a particular state and false in others, while a proposition such as "A implies A" should be true for all propositions A in all states. In linear logic we can capture this with

$$\frac{\cdot \vdash A\ \mathit{true}}{\vdash A\ \mathit{valid}}$$

It expresses that if A is true without using any hypotheses, then A is valid.

From the perspective of linear logic, A being valid means we should be able to produce as many proofs of A as we wish. This in turn allows us to use A as many times as we wish in a proof. After all, a proof of A requires no resources. Using cut:

$$\frac{\cdot \vdash A \quad \Delta, A \vdash C}{\Delta \vdash C}\ \mathsf{cut}$$

So we should treat *antecedents* $A\ \mathit{valid}$ as *structural*.

The judgment we end up with has the form anticipated at the beginning of this section.

$$\underbrace{\Gamma}_{\mathit{valid}} \,;\, \underbrace{\Delta}_{\mathit{true}} \vdash A\ \mathit{true}$$

An interesting property of this formulation is that we do not change (yet) our language of propositions at all: they are all linear, with the usual linear connectives. This means that propositional inference rules are applied only to the succedent $A\ \mathit{true}$ and the antecedents in Δ, not those in Γ.

Once we have the judgment $A\ \mathit{valid}$ we also obtain a second judgment

$$\underbrace{\Gamma}_{\mathit{valid}} \vdash A\ \mathit{valid}$$

but there is only a single rule that applies here because the meaning of the connectives arises entirely from their linear nature. An antecedent $A\ \mathit{valid}$ allows us to use it whenever we wish.

$$\frac{\Gamma\ ;\ \cdot \vdash A\ \mathit{true}}{\Gamma \vdash A\ \mathit{valid}}\ \mathsf{valid}_R \qquad \frac{\Gamma, A\ \mathit{valid}\ ;\ \Delta, A\ \mathit{true} \vdash C\ \mathit{true}}{\Gamma, A\ \mathit{valid}\ ;\ \Delta \vdash C\ \mathit{true}}\ \mathsf{valid}_L$$

These rules are not regular right or left rules, because validity is a judgment, not a proposition. Just to remind ourselves of this we write R and L as a subscript.

In some ways these rules are similar to cut and identity in the sense that they apply to arbitrary propositions A. So rather than defining the nature of the connectives, they define the nature of the judgments. Cut and identity explain the nature of the hypothetical judgment, while valid_R and valid_L explain the nature of validity.

The next question is about how to express validity, internally, as a proposition. At this point this has become easy!

$$\frac{\Gamma \vdash A\ \mathit{valid}}{\Gamma\ ;\ \cdot \vdash\ !A\ \mathit{true}}\ !R \qquad \frac{\Gamma, A\ \mathit{valid}\ ;\ \Delta \vdash C\ \mathit{true}}{\Gamma\ ;\ \Delta, !A \vdash C\ \mathit{true}}\ !L$$

Since there is only one rule to conclude $\Gamma \vdash A\ \mathit{valid}$ (namely valid_R) the system is a little less symmetric but a little more streamlined if we combine $!R$ with valid_R into the rule

$$\frac{\Gamma\ ;\ \cdot \vdash A\ \mathit{true}}{\Gamma\ ;\ \cdot \vdash\ !A\ \mathit{true}}\ !R'$$

As discussed in lecture, we couldn't allow a nonempty Δ in the conclusion of $!R'$ (or $!R$, for that matter) because in the premise it must definitely be empty, and then none of the supposedly linear antecedents in Δ would actually be used.

As one might expect, things *do* go horribly wrong without this restriction. Since Γ is structural, it is just added parametrically to all the right and left rules and we have and also allowed for cut and identity.

$$\frac{}{\Gamma\ ;\ A \vdash A}\ \mathsf{id} \qquad \frac{\Gamma\ ;\ \Delta \vdash A \quad \Gamma\ ;\ \Delta' \vdash C}{\Gamma\ ;\ \Delta, \Delta' \vdash C}\ \mathsf{cut}$$

Without the restriction on $!R'$, we could prove:

$$\dfrac{\dfrac{\dfrac{}{\cdot\ ;\ A \vdash A}\ \mathsf{id}}{\cdot\ ;\ A \vdash\ !A}\ {!R'??} \qquad \dfrac{\dfrac{\dfrac{\dfrac{}{A\ ;\ A \vdash A}\ \mathsf{id}}{A\ ;\ \cdot \vdash A}\ \mathsf{valid}_L \quad \dfrac{\dfrac{}{A\ ;\ A \vdash A}\ \mathsf{id}}{A\ ;\ \cdot \vdash A}\ \mathsf{valid}_L}{A\ ;\ \cdot \vdash A \otimes A}\ \otimes R}{\cdot\ ;\ !A \vdash A \otimes A}\ !L}{\cdot\ ;\ A \vdash A \otimes A}\ \mathsf{cut}$$

and similarly $\cdot\ ;\ A \vdash 1$. For these it is an easy step to show that weakening and contraction for all *linear* antecedents would be admissible. In other words, the logic would no longer be linear!

4 Examples

As mentioned in the introduction, the construction of validity is quite generic. For example, it could be carried out even if the base logic were already structural, in which case we obtain a version of the intuitionistic modal logic S4 [Pfenning and Davies, 2001], where $!A$ would be written $\Box A$. So we can test some laws of modal logic here. The first three judgments below are derivable; the last one isn't.

$$\begin{array}{l} \vdash\ !(A \multimap B) \multimap (!A \multimap !B) \\ \vdash\ !A \multimap A \\ \vdash\ !A \multimap !!A \\ \nvdash P \multimap !P \end{array}$$

Let's write out the first one, which indicates that linear logic is a "normal" modal logic because the exponential distributes over implication.

$$\dfrac{\begin{array}{c}\vdots \\ \cdot\ ;\ !(A \multimap B), !A \vdash\ !B\end{array}}{\cdot\ ;\ \cdot \vdash\ !(A \multimap B) \multimap (!A \multimap !B)}\ {\multimap}R \times 2$$

At this point we cannot apply $!R$ because there are linear antecedents, so we have to shuffle them into the structural antecedents and then apply $!R'$.

$$\dfrac{\dfrac{\dfrac{\begin{array}{c}\vdots \\ A \multimap B, A\ ;\ \cdot \vdash B\end{array}}{A \multimap B, A\ ;\ \cdot \vdash\ !B}\ !R'}{\cdot\ ;\ !(A \multimap B), !A \vdash\ !B}\ !L \times 2}{\cdot\ ;\ \cdot \vdash\ !(A \multimap B) \multimap (!A \multimap !B)}\ {\multimap}R \times 2$$

Now we can copy $A \multimap B$ to the linear context since we would like to apply a left rule to it. The premises of ${\multimap}L$ then follow readily.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{A \multimap B, A\ ;\ A \vdash A}\ \mathsf{id}}{A \multimap B, A\ ;\ \cdot \vdash A}\ \mathsf{valid}_L \quad \dfrac{}{A \multimap B, A\ ;\ B \vdash B}\ \mathsf{id}}{A \multimap B, A\ ;\ A \multimap B \vdash B}\ {\multimap}L}{A \multimap B, A\ ;\ \cdot \vdash B}\ \mathsf{valid}_L}{A \multimap B, A\ ;\ \cdot \vdash\ !B}\ !R'}{\cdot\ ;\ !(A \multimap B), !A \vdash\ !B}\ !L \times 2}{\cdot\ ;\ \cdot \vdash\ !(A \multimap B) \multimap (!A \multimap !B)}\ {\multimap}R \times 2$$

This illustrates a practical shortcoming of this system: since right and left rules are applied only to linear propositions, we frequently have to move structural propositions into and out of the linear antecedents, using valid_L and $!L$.

We can consider other questions. For example, does the exponential distribute over the tensor? Let's try:

$$\dfrac{\begin{matrix}\vdots \\ A \otimes B \,;\, \cdot \vdash {!A} \otimes {!B}\end{matrix}}{\cdot \,;\, !(A \otimes B) \vdash {!A} \otimes {!B}}\;!L$$

At this point we could try $\otimes R$ or valid_L. After $\otimes R$ there are two remaining symmetric subgoals.

$$\dfrac{\dfrac{\dfrac{\begin{matrix}\text{fails} \\ A \otimes B \,;\, \cdot \vdash A\end{matrix}}{A \otimes B \,;\, \cdot \vdash {!A}}\;!R' \qquad \begin{matrix}\vdots \\ A \otimes B \,;\, \cdot \vdash {!B}\end{matrix}}{A \otimes B \,;\, \cdot \vdash {!A} \otimes {!B}}\;\otimes R}{\cdot \,;\, !(A \otimes B) \vdash {!A} \otimes {!B}}\;!L$$

We can prove neither of them, because whenever we copy $A \otimes B$ into the linear zone, followed by $\otimes L$, we get both A and B, linearly, but we only have A to prove.

If we try valid_L first, we also get stuck because we have linear A and B but the ultimately succedents are $!A$ and $!B$.

$$\dfrac{\dfrac{\dfrac{\dfrac{\begin{matrix}\text{fails} \\ A \otimes B \,;\, A \vdash {!A}\end{matrix} \qquad \begin{matrix}\text{fails} \\ A \otimes B \,;\, B \vdash {!B}\end{matrix}}{A \otimes B \,;\, A, B \vdash {!A} \otimes {!B}}\;\otimes R}{A \otimes B \,;\, A \otimes B \vdash {!A} \otimes {!B}}\;\otimes L}{A \otimes B \,;\, \cdot \vdash {!A} \otimes {!B}}\;\mathsf{valid}_L}{\cdot \,;\, !(A \otimes B) \vdash {!A} \otimes {!B}}\;!L$$

The failure of these attempts doesn't mean much, but since this logic satisfies cut and identity elimination (see Section 6) it takes just a little more work to show that these are in fact not provable.

Perhaps we should have even seen intuitively that this entailment does not hold. It says that if we have both A and B together, as many times as we want, we can get, independently, A as often as we want and B as often as we want. That just couldn't be true.

But the exponential distributes over the additive conjunction $A \,\&\, B$ in an interesting way. Intuitively, $!(A \,\&\, B)$ means that we arbitrarily often have a choice between A and B. and $!A \otimes !B$ means that we have both A and B arbitrarily often, separately. These are equivalent.

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{A \mathbin{\&} B \,;\, A \vdash A}\,\mathsf{id}}{A \mathbin{\&} B \,;\, A \mathbin{\&} B \vdash A}\,\&L_1}{A \mathbin{\&} B \,;\, \cdot \vdash A}\,\mathsf{valid}_L}{A \mathbin{\&} B \,;\, \cdot \vdash {!A}}\,{!R'} \qquad \dfrac{\dfrac{\dfrac{\dfrac{}{A \mathbin{\&} B \,;\, B \vdash B}\,\mathsf{id}}{A \mathbin{\&} B \,;\, A \mathbin{\&} B \vdash B}\,\&L_2}{A \mathbin{\&} B \,;\, \cdot \vdash B}\,\mathsf{valid}_L}{A \mathbin{\&} B \,;\, \cdot \vdash {!B}}\,{!R'}}{\dfrac{A \mathbin{\&} B \,;\, \cdot \vdash {!A} \otimes {!B}}{\cdot \,;\, {!(A \mathbin{\&} B)} \vdash {!A} \otimes {!B}}\,{!L}}\,\otimes R
$$

The other direction, $\cdot \,;\, {!A} \otimes {!B} \vdash {!(A \mathbin{\&} B)}$ is perhaps even more straightforward.

5 Translation from Structural into Linear Logic

The whole endeavor of linear logic is to add expressive power to structural logic. This is clearly not the case without either recursion (which jeopardizes the logical reading altogether) or the exponential. Now that we have validity (which is structural) and the exponential modality, how do we translate ordinary intuitionistic logic into linear logic?

There seem to be fundamentally two translations, one "by name" and one "by value". They are so named because of what they mean operationally, under a functional interpretation. Girard [1987] provides a "by value" translation, so we develop that.

The basic idea guiding the translation $(A)^\vee$ is that if $\Gamma \vdash A$ then $(\Gamma)^\vee \,;\, \cdot \vdash (A)^\vee$. Using one of the judgments we have introduced, we could also have said $(\Gamma)^\vee \vdash (A)^\vee\ \mathit{valid}$. This should not be so surprising. The other direction is generally easy because we can just ignore the strictures of linearity.

Now we examine a few connectives in turn to see how they should translate.

$$
\dfrac{\Gamma, A \vdash B}{\Gamma \vdash A \supset B}\,{\supset}R \qquad\qquad \begin{array}{c} \Gamma^\vee, A^\vee \,;\, \cdot \vdash B^\vee \\ \vdots \\ \Gamma^\vee \,;\, \cdot \vdash (A \supset B)^\vee \end{array}
$$

We see that at least for the right rule, we can pick

$$
(A \supset B)^\vee = {!A^\vee} \multimap {!B^\vee}
$$

and then apply $!L$ and $!R'$ after $\multimap R$.

$$
\dfrac{\Gamma, A \vdash B}{\Gamma \vdash A \supset B}\,{\supset}R \qquad\qquad \dfrac{\dfrac{\dfrac{\Gamma^\vee, A^\vee \,;\, \cdot \vdash B^\vee}{\Gamma^\vee, A^\vee \,;\, \cdot \vdash {!B^\vee}}\,{!R'}}{\Gamma^\vee \,;\, {!A^\vee} \vdash {!B^\vee}}\,{!L}}{\Gamma^\vee \,;\, \cdot \vdash {!A^\vee} \supset {!B^\vee}}\,{\multimap}R
$$

What about the left rule?

$$\frac{\Gamma, A \supset B \vdash A \quad \Gamma, A \supset B, B \vdash C}{\Gamma, A \supset B \vdash C} \supset L \qquad \begin{array}{c} \Gamma^\vee, !A^\vee \multimap !B^\vee \,;\, \cdot \vdash A^\vee \quad \Gamma^\vee, !A^\vee \multimap !B^\vee, B^\vee \,;\, \cdot \vdash C^\vee \\ \vdots \\ \Gamma^\vee, !A^\vee \multimap !B^\vee \,;\, \cdot \vdash C^\vee \end{array}$$

Note that because Γ and $\Gamma^\vee$ are both structural, we propagate them to all premises. Then we can just copy $!A^\vee \multimap !B^\vee$ to the linear antecedent, and apply the left rule, followed by some more administrative moves.

$$\cfrac{\cfrac{\cfrac{\Gamma^\vee, !A^\vee \multimap !B^\vee \,;\, \cdot \vdash A^\vee}{\Gamma^\vee, !A^\vee \multimap !B^\vee \,;\, \cdot \vdash !A^\vee} !R \quad \cfrac{\Gamma^\vee, !A^\vee \multimap !B^\vee, B^\vee \,;\, \cdot \vdash C^\vee}{\Gamma^\vee, !A^\vee \multimap !B^\vee \,;\, !B^\vee \vdash C^\vee} !L}{\Gamma^\vee, !A^\vee \multimap !B^\vee \,;\, !A^\vee \multimap !B^\vee \vdash C^\vee} \multimap L}{\Gamma^\vee, !A^\vee \multimap !B^\vee \,;\, \cdot \vdash C^\vee} \mathsf{valid}L$$

All the other connectives follow similar patterns, but let's also look at identity and cut.

$$\frac{}{\Gamma, A \vdash A} \mathsf{id} \qquad \cfrac{\cfrac{}{\Gamma^\vee, A^\vee \,;\, A^\vee \vdash A^\vee} \mathsf{id}}{\Gamma^\vee, A^\vee \,;\, \cdot \vdash A^\vee} \mathsf{valid}_L$$

$$\frac{\Gamma \vdash A \quad \Gamma, A \vdash C}{\Gamma \vdash C} \mathsf{cut} \qquad \begin{array}{c} \Gamma^\vee \,;\, \cdot \vdash A^\vee \quad \Gamma^\vee, A^\vee \,;\, \cdot \vdash C^\vee \\ \vdots \\ \Gamma^\vee \,;\, \cdot \vdash C^\vee \end{array}$$

We can derive the conclusion in the case of a cut by introducing and then cutting $!A^\vee$

$$\cfrac{\cfrac{\Gamma^\vee \,;\, \cdot \vdash A^\vee}{\Gamma^\vee \,;\, \cdot \vdash !A^\vee} !R' \quad \cfrac{\Gamma^\vee, A^\vee \,;\, \cdot \vdash C^\vee}{\Gamma^\vee \,;\, !A^\vee \vdash C^\vee} !L}{\Gamma^\vee \,;\, \cdot \vdash C^\vee} \mathsf{cut}$$

As we will see in Section 6 is also makes sense to extend linear logic with additional rule that cuts valid antecedents directly.

$$\frac{\Gamma^\vee \,;\, \cdot \vdash A^\vee \quad \Gamma^\vee, A^\vee \,;\, \cdot \vdash C^\vee}{\Gamma^\vee \,;\, \cdot \vdash C^\vee} \mathsf{cut!}$$

To completing the translation we map the intuitionistic (structural) connectives to their linear counterparts and prefix every subformula with an exponential. In the reverse direction $A^\wedge$ we just map all linear connectives to their structural counterparts and drop all exponentials.

$$
\begin{array}{lcl@{\qquad}lcl}
(A \supset B)^\vee & = & !A^\vee \multimap !B^\vee & (A \multimap B)^\wedge & = & A^\wedge \supset B^\wedge \\
(A \wedge B)^\vee & = & !A^\vee \otimes !B^\vee & (A \otimes B)^\wedge & = & A^\wedge \wedge B^\wedge \\
 & & & (A \mathbin{\&} B)^\wedge & = & A^\wedge \wedge B^\wedge \\
(\top)^\vee & = & \mathbf{1} & (\mathbf{1})^\wedge & = & \top \\
 & & & (\top)^\wedge & = & \top \\
(A \vee B)^\vee & = & !A^\vee \oplus !B^\vee & (A \oplus B)^\wedge & = & A^\wedge \vee B^\wedge \\
(\bot)^\vee & = & \mathbf{0} & (\mathbf{0})^\wedge & = & \bot \\
 & & & (!A)^\wedge & = & A^\wedge \\
(P)^\vee & = & P & (P)^\wedge & = & P
\end{array}
$$

We can summarize the correctness of the translation in the following theorem.

Theorem 1 (Correctness of Translation from Structural to Linear Logic)

(i) *If* $\Gamma \vdash A$ *then* $\Gamma^\vee \,;\, \cdot \vdash A^\vee$

(ii) *If* $\Gamma \,;\, \Delta \vdash A$ *then* $\Gamma^\wedge, \Delta^\wedge \vdash A^\wedge$

Proof: Part (i) follows by structural induction over the given derivation. In each case we directly construct the resulting derivation, preserving the essential structure while inserting rules concerning validity and the exponential. We showed some representative cases in this section.

Part (ii) also follows by structural induction over the given derivation. Some structural antecedents available for $\Gamma^\wedge, \Delta^\wedge \vdash A^\wedge$ will be unnecessary and can be dropped by weakening. □

There is an optimized translation where the subformulas of positive propositions ($\otimes$, $\oplus$) are not preceded by an exponential. I suspect the most straightforward way to prove the correctness of the optimized translation is to prove inversion of the left rules on the structural side and then mimic them with the linear left rules (which also happen to be invertible).

6 Cut and Identity Elimination[1]

Both cut and identity elimination carry over from the purely linear case, but with a few new wrinkles.

[1] not covered in lecture

Theorem 2 (Admissibility of Identity) *If we restrict the identity to atomic propositions, then*

$$\frac{}{\Gamma \, ; A \vdash A}\ \mathsf{id}_A$$

is admissible for arbitrary A.

Proof: As before, by structural induction on A. The only interesting case is $A = !A'$. We construct:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\Gamma, A' \, ; A' \vdash A'}\ \mathsf{id}_{A'}}{\Gamma, A' \, ; \cdot \vdash A'}\ \mathsf{valid}_L}{\Gamma, A' \, ; \cdot \vdash !A'}\ !R}{\Gamma \, ; !A' \vdash !A'}\ !L$$

□

In order to prove admissibility of cut, it is helpful to simultaneously proof the admissibility of cut!. The induction measure is then somewhat more complicated, as we explain below. The first premise of the cut! rule expresses that A is valid, so we can cut out an antecedent of the form A *valid* from the second premise.

Theorem 3 (Admissibility of Cut) *The rules*

$$\frac{\Gamma \, ; \Delta \vdash A \quad \Gamma \, ; \Delta' \vdash C}{\Gamma \, ; \Delta, \Delta' \vdash C}\ \mathsf{cut} \qquad \frac{\Gamma \, ; \cdot \vdash A \quad \Gamma, A \, ; \Delta' \vdash C}{\Gamma \, ; \Delta' \vdash C}\ \mathsf{cut!}$$

are admissible.

Proof: By a simultaneous nested induction in the following order

(1) the structure of the proposition A

(2) $\mathsf{cut!}_A$ is greater than cut_A

(3) either the first or the second derivation becomes smaller while the other remains the same

Item (2) is new here and necessary for the following case.

Case:

$$\dfrac{\begin{array}{c}\mathcal{D} \\ \Gamma \, ; \cdot \vdash A\end{array} \quad \dfrac{\begin{array}{c}\mathcal{E}' \\ \Gamma, A \, ; \Delta', A \vdash C\end{array}}{\Gamma, A \, ; \Delta' \vdash C}\ \mathsf{valid}_L}{\Gamma \, ; \Delta' \vdash C}\ \mathsf{cut!}_A$$

We have to construct a new derivation with two cuts, because there are now two copies of A among the antecedents of $\mathcal{E}'$.

$$\dfrac{\begin{array}{c}\mathcal{D}\\ \Gamma\,;\cdot\vdash A\end{array} \qquad \dfrac{\begin{array}{c}\mathcal{D}\\ \Gamma\,;\cdot\vdash A\end{array} \quad \begin{array}{c}\mathcal{E}'\\ \Gamma, A\,;\Delta', A\vdash C\end{array}}{\Gamma\,;\Delta', A\vdash C}\ \mathsf{cut!}_A}{\Gamma\,;\Delta'\vdash C}\ \mathsf{cut}_A$$

The problem here is that both cut_A and $\mathsf{cut!}_A$ are on the same cut formula A. Also, the derivation of the second premise of cut_A may be much larger than the original $\mathcal{E}$, since it is the result of the induction hypothesis on $\mathsf{cut!}_A$. So we need that cut_A is strictly smaller than $\mathsf{cut!}_A$. Fortunately, the other critical case (which necessitates $\mathsf{cut!}_A$ in the first place) requires an appeal to the induction hypothesis at a smaller proposition.

Case:

$$\dfrac{\dfrac{\begin{array}{c}\mathcal{D}'\\ \Gamma\,;\cdot\vdash A'\end{array}}{\Gamma\,;\cdot\vdash {!A'}}\ !R \qquad \dfrac{\begin{array}{c}\mathcal{E}'\\ \Gamma, A'\,;\Delta'\vdash C\end{array}}{\Gamma\,;\Delta', {!A'}\vdash C}\ !L}{\Gamma\,;\Delta'\vdash C}\ \mathsf{cut}_{!A'}$$

We reduce this immediately to a $\mathsf{cut!}_{A'}$, which is a smaller formula. So even if $\mathsf{cut!}_A$ is greater than cut_A, the structure of the proposition takes precedence.

$$\dfrac{\begin{array}{c}\mathcal{D}'\\ \Gamma\,;\cdot\vdash A'\end{array} \quad \begin{array}{c}\mathcal{E}'\\ \Gamma, A'\,;\Delta'\vdash C\end{array}}{\Gamma\,;\Delta'\vdash C}\ \mathsf{cut!}_{A'}$$

□

References

Jean-Marc Andreoli. Logic programming with focusing proofs in linear logic. *Journal of Logic and Computation*, 2(3):197–347, 1992.

Andrew Barber. Dual intuitionistic linear logic. Technical Report ECS-LFCS-96-347, Department of Computer Science, University of Edinburgh, September 1996.

Jean-Yves Girard. Linear logic. *Theoretical Computer Science*, 50:1–102, 1987.

Jean-Yves Girard and Yves Lafont. Linear logic and lazy computation. In H. Ehrig, R. Kowalski, G. Levi, and U. Montanari, editors, *Proceedings of the International Joint Conference on Theory and Practice of Software Development*, volume 2, pages 52–66, Pisa, Italy, March 1987. Springer-Verlag LNCS 250.

Frank Pfenning and Rowan Davies. A judgmental reconstruction of modal logic. *Mathematical Structures in Computer Science*, 11:511–540, 2001. Notes to an invited talk at the *Workshop on Intuitionistic Modal Logics and Applications* (IMLA'99), Trento, Italy, July 1999.