Available online at www.sciencedirect.com

ScienceDirect

Electronic Notes in Theoretical Computer Science 312 (2015) 107–124

Electronic Notes in Theoretical Computer Science

www.elsevier.com/locate/entcs

Yet Another Bijection Between Sequent Calculus and Natural Deduction [1]

Cecilia Englander [2]

Departmento de Informática
PUC-Rio
Rio de Janeiro, Brazil

Gilles Dowek [3]

Inria
Paris, France

Edward Hermann Haeusler [4]

Departmento de Informática
PUC-Rio
Rio de Janeiro, Brazil

Abstract

This work shows a bijection between sequent calculus and natural deduction for intuitionistic propositional logic so far as normal and cut-free derivations are concerned.

Keywords: Correspondence, Natural Deduction, Sequent Calculus, Intuitionistic propositional logic, Proof Theory

1 Introduction

Equivalences between natural deduction and sequent calculus have been discussed since their definition by Gentzen [2]. By equivalence between the systems we mean that every derivation in one system can be transformed into a derivation in the other. Such equivalence being established, the search for a stronger equivalence starts. Some examples are Zucker [13], who shows a correspondence between normalization

[1] The authors are grateful to CAPES and CNPq for their financial support.

[2] Email: `cenglander@inf.puc-rio.br`

[3] Email: `gilles.dowek@inria.fr`

[4] Email: `hermann@inf.puc-rio.br`

http://dx.doi.org/10.1016/j.entcs.2015.04.007

1571-0661/© 2015 Published by Elsevier B.V.

and cut-elimination for the fragment $\{\wedge, \rightarrow, \forall, \bot\}$, followed by Pottinger [11], who improved Zucker's method by simplifying it and extending it to the full intuitionistic propositional logic. Danos, Joinet and Schellinx [1] have an isomorphism between Sequent Calculus and Natural Deduction passing through Linear Logic. Nigam and Miller [10] showed that different proof systems, including Natural Deduction and Sequent Calculus, have the same provable sets of formulas by encoding the systems into a Focused Linear Logic. In [4], Henriksen showed that Linear Logic is not needed and showed a similar result from that of [10] by encoding the systems into a focused intuitionistic system. Negri and von Plato [9] showed the relation between structural rules in sequent calculus and discharge of formulas in natural deduction. Due to the structural rules, the correspondence shown in [9] is not one-to-one. For example, a derivation of the implication $A \rightarrow (B \rightarrow (B \rightarrow (A \rightarrow B)))$ in Sequent Calculus has three applications of the weakening rule and these applications can appear in the beginning of the derivation (in different orders) or in different levels of the derivations. Two possible derivations for this implication, with the notation used in [9], are:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{B \Rightarrow B}{A, B \Rightarrow B}\,Wk}{A, A, B \Rightarrow B}\,Wk}{A, A, B, B \Rightarrow B}\,Wk}{A, B, B \Rightarrow A \supset B}\,R\supset}{A, B \Rightarrow B \supset (A \supset B)}\,R\supset}{A \Rightarrow B \supset (B \supset (A \supset B))}\,R\supset}{\Rightarrow A \supset (B \supset (B \supset (A \supset B)))}\,R\supset
\qquad
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{B \Rightarrow B}{A, B \Rightarrow B}\,Wk}{B \Rightarrow A \supset B}\,R\supset}{\Rightarrow B \supset (A \supset B)}\,R\supset}{B \Rightarrow B \supset (A \supset B)}\,Wk}{\Rightarrow B \supset (B \supset (A \supset B))}\,R\supset}{A \Rightarrow B \supset (B \supset (A \supset B))}\,Wk}{\Rightarrow A \supset (B \supset (B \supset (A \supset B)))}\,R\supset$$

The derivation on the left side has no correspondent in Natural Deduction and the derivation on the right side corresponds to the derivation

$$\dfrac{\dfrac{\dfrac{\dfrac{\overset{2.}{[B]}}{A \supset B}\,I\supset, 1.}{B \rightarrow (A \supset B)}\,I\supset, 2.}{B \supset (B \rightarrow (A \supset B))}\,I\supset, 3.}{A \supset (B \supset (B \rightarrow (A \supset B)))}\,I\supset, 4.$$

in Natural Deduction, where 1., 3., and 4., are "ghost" labels that correspond to vacuous discharge. In the systems we are going to work with there is only one possible (normal/cut-free) derivation of $A \rightarrow (B \rightarrow (B \rightarrow (A \rightarrow B)))$ in Sequent Calculus and in Natural Deduction.

There are more proofs of a proposition in sequent calculus than in natural deduction. For instance, there are two possible cut-free derivations for the proposition $(A \wedge B) \rightarrow (A \vee C)$ in the sequent calculus system defined in [3]:

$$\dfrac{\dfrac{\dfrac{A \vdash A}{A \vdash A \vee C}\mathcal{R}1\vee}{A \wedge B \vdash A \vee C}\mathcal{L}1\wedge}{\vdash (A \wedge B) \to (A \vee C)}\mathcal{R}\Rightarrow \qquad \dfrac{\dfrac{\dfrac{A \vdash A}{A \wedge B \vdash A}\mathcal{L}1\wedge}{A \wedge B \vdash A \vee C}\mathcal{R}1\vee}{\vdash (A \wedge B) \to (A \vee C)}\mathcal{R}\Rightarrow$$

and only one in the natural deduction system defined in [12]:

$$\dfrac{\dfrac{\dfrac{\overset{(1)}{A\&B}}{A}}{A \vee C}}{(A\&B) \to (A \vee C)}\,(1)$$

Thus, to define an isomorphism, we need to choose a more restrictive sequent calculus and/or a more liberal natural deduction system.

Our goal is to show that the relation between cut-free and normal derivations are stronger than what is shown in [11,9,7].

1.1 Sequent Calculus

LJT is the implicational fragment of LKT which was first introduced in Joinet's thesis [7]. In fact, the system introduced by Joinet is a slight different version of LJT, called ILU to stress that this fragment of LKT could also be seen as the intuitionistic fragment of Girard's LU.

In [5], Herbelin defined an extension of the usual λ-calculus called $\overline{\lambda}$-calculus. However, for a λ-term that corresponds to a $\overline{\lambda}$-term of the form $(\ldots(x[u_1])\ldots[u_k])$, the LJT image is a proof with cuts. This term is a $\overline{\lambda}$ image of the normal term $(\ldots(x\ u_1)\ldots u_n)$ [5], but in $\overline{\lambda}$ it is not normal due to the use of explicit substitution in $\overline{\lambda}$. Thus, [5] reports a mapping between λ and LJT that takes normal terms as those shown in λ into derivations in LJT with cuts. In our proposed isomorphism, we avoid this by using a notion of proof equivalence and different versions of sequent calculus and natural deduction. The paper [5] only deals with the implicational fragment of intuitionistic logic, but in his thesis [6], Herbelin extends the result to the full propositional fragment of intuitionistic logic.

In table 1 we present LJT for the full intuitionistic propositional fragment $\{\wedge, \vee, \to, \neg, \perp\}$, where negation ($\neg$) can be seen as a particular case of implication in which the consequence is always a falsity.

A sequent in LJT is of the form $\Gamma; \Delta \vdash \gamma$, where Γ is a set of formulas (possibly empty), γ is a formula and Δ is a set of at most one formula. The place occupied by Δ is called *stoup* and the formula in the stoup (if any) is called *head-formula.* In a derivation, the stoup of the bottomost sequent must be empty.

Herbelin's version of LJT is a slightly different version of the intuitionistic fragment of LKT which was first introduced in Joinet's PhD thesis [7]. The differences are: (1) the formulas that form the disjunction in $\vee\vdash$ are outside the stoup, (2) to apply the right rules the stoup must be empty and (3) the rule $\mathcal{D}$ keeps a copy of

[5] $(\ldots(x\ u_1)\ldots u_n)$ is normal in λ whenever u_i is normal.

$$(\to\vdash)\ \frac{\Gamma;\vdash\alpha \quad \Gamma;\beta\vdash\gamma}{\Gamma;\alpha\to\beta\vdash\gamma} \qquad (\vdash\to)\ \frac{\Gamma,\alpha;\vdash\beta}{\Gamma;\vdash\alpha\to\beta}$$

$$(\wedge\vdash)\ \frac{\Gamma;\beta\vdash\gamma}{\Gamma;\alpha\wedge\beta\vdash\gamma} \quad \frac{\Gamma;\alpha\vdash\gamma}{\Gamma;\alpha\wedge\beta\vdash\gamma} \qquad (\vdash\wedge)\ \frac{\Gamma;\vdash\alpha \quad \Gamma;\vdash\beta}{\Gamma;\vdash\alpha\wedge\beta}$$

$$(\vee\vdash)\ \frac{\Gamma,\alpha;\vdash\gamma \quad \Gamma,\beta;\vdash\gamma}{\Gamma;\alpha\vee\beta\vdash\gamma} \qquad (\vdash\vee)\ \frac{\Gamma;\vdash\alpha}{\Gamma;\vdash\alpha\vee\beta} \quad \frac{\Gamma;\vdash\beta}{\Gamma;\vdash\alpha\vee\beta}$$

$$(\bot\vdash)\ \frac{\Gamma;\alpha\vdash\gamma}{\Gamma;\bot\vdash\gamma}$$

$$Ax\ \frac{}{\Gamma;\alpha\vdash\alpha} \qquad \mathcal{D}\ \frac{\Gamma,\alpha;\alpha\vdash\gamma}{\Gamma,\alpha;\vdash\gamma}$$

Table 1
LJT rules

the formula that passed to the stoup (4) the left rules for conjunction, which are three:

$$\frac{\Gamma,\alpha;\beta\vdash\gamma}{\Gamma;\alpha\wedge\beta\vdash\gamma} \qquad \frac{\Gamma,\beta;\alpha\vdash\gamma}{\Gamma;\alpha\wedge\beta\vdash\gamma} \qquad \frac{\Gamma,\beta,\alpha;\vdash\gamma}{\Gamma;\alpha\wedge\beta\vdash\gamma}$$

The version of LJT presented here is Herbelin's version, except for the rules of left conjunction, which are like in Joinet's thesis.

An inference rule can be read from the conclusion to its premises. Note that we can only apply right rules when the stoup is empty and that we can bring a formula to the stoup with the rule $\mathcal{D}$ but we cannot take a formula from the stoup. LJT has additive contexts, i.e., the same set Γ of assumptions in the premises of each rule, and even though we need $\mathcal{D}$ as a rule in the system.

LJT forces a focusing in the derivation. When there is a formula in the stoup, we are "forced" to apply left rules, breaking the named formula until either an atomic formula is in the stoup, in which case we have an initial sequent (that is, the topmost sequent in a derivation), or until we apply $\vee\vdash$, in which case the stoup is empty, and we can choose between applying a right rule and the rule $\mathcal{D}$, in which case the focus is back to the head-formula. When the bottommost rule applied in a cut-free derivation is a $\mathcal{D}$-rule, we can identify the trunk of applications of left rules forced by the stoup with a positive trunk in focused proofs[6].

Due to the stoup, the system admits two cuts, a *head-cut* (C_H) which cuts the formula in the stoup and a *middle-cut* (C_M) which cuts a formula outside the stoup:

$$\frac{\Gamma;\Delta\vdash A \quad \Gamma;A\vdash B}{\Gamma;\Delta\vdash B}\,C_H \qquad \frac{\Gamma;\vdash A \quad \Gamma,A;\Delta\vdash B}{\Gamma;\Delta\vdash B}\,C_M$$

[6] This terminology is according to [8]

Definition 1.1 [Cut-free derivation] We say that a derivation Π is *cut-free* in LJT when there is neither applications of C_H nor applications of C_M in Π.

The stoup, that is, the use of focusing, reduces the amount of derivations that we usually have in sequent calculus. For instance, instead of the two possible cut-free sequent calculus derivations of $(A \wedge B) \to (A \vee C)$ we showed in last section, we only have one in LJT (see figure 1).

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{A \wedge B; A \vdash A}\,Ax}{A \wedge B; A \wedge B \vdash A}\,\wedge\vdash}{A \wedge B; \vdash A}\,\mathcal{D}}{A \wedge B; \vdash A \vee C}\,\vdash\vee}{; \vdash (A \wedge B) \to (A \vee C)}\,\vdash\to$$

Fig. 1. Example of a derivation in LJT

1.2 Natural Deduction

If we decide to use Gentzen's natural deduction system $\mathcal{N}J$, there would be no way to distinguish derivations with more premises than needed. For instance, derivations of $A \wedge B$ from $A \wedge B$ and of $A \wedge B$ from $A \wedge B, C$ in LJT would be translated to the same derivation in $\mathcal{N}J$.

In order to have a faithful comparison, we decided to represent natural deduction in a sequent calculus style. The system ND is presented in table 2 and the rules can also be read bottom-up.

$$(E_\to)\ \dfrac{\Gamma \vdash \alpha \to \beta \quad \Gamma \vdash \alpha}{\Gamma \vdash \beta} \qquad\qquad (I_\to)\ \dfrac{\Gamma, \alpha \vdash \beta}{\Gamma \vdash \alpha \to \beta}$$

$$(E_\wedge)\ \dfrac{\Gamma \vdash \alpha \wedge \beta}{\Gamma \vdash \alpha} \qquad \dfrac{\Gamma \vdash \alpha \wedge \beta}{\Gamma \vdash \beta} \qquad\qquad (I_\wedge)\ \dfrac{\Gamma \vdash \alpha \quad \Gamma \vdash \beta}{\Gamma \vdash \alpha \wedge \beta}$$

$$(E_\vee)\ \dfrac{\Gamma \vdash \alpha \vee \beta \quad \Gamma, \alpha \vdash \gamma \quad \Gamma, \beta \vdash \gamma}{\Gamma \vdash \gamma} \qquad\qquad (I_\vee)\ \dfrac{\Gamma \vdash \alpha}{\Gamma \vdash \alpha \vee \beta} \qquad \dfrac{\Gamma \vdash \beta}{\Gamma \vdash \alpha \vee \beta}$$

$$(E_\perp)\ \dfrac{\Gamma \vdash \perp}{\Gamma \vdash \gamma} \qquad\qquad Ax\ \dfrac{}{\Gamma, \alpha \vdash \alpha}$$

Table 2
ND rules

Definition 1.2 [Major premise] The premises $\Gamma \vdash \alpha \to \beta$, $\Gamma \vdash \alpha \wedge \beta$, $\Gamma \vdash \alpha \vee \beta$ and $\Gamma \vdash \perp$ are the *major premises* of the rules $E_\to$, $E_\wedge$, $E_\vee$ and $E_\perp$, respectively. The other premises are called *minor premises*.

A cut rule in sequent calculus is usually mapped into a derivation with a *maximal sequent* in natural deduction, that is, a sequent which is the conclusion of an introduction rule and major premise of an elimination rule. But LJT has two cuts: if one is translated to a derivation with a maximal sequent, what would the other cut represent in natural deduction? We add to our system the following admissible rule known as *substitution rule*:

$$\frac{\Gamma \vdash \alpha \quad \Gamma, \alpha \vdash \beta}{\Gamma \vdash \beta}\ \mathcal{S}$$

Definition 1.3 [Normal derivation] A derivation Π is *normal* in ND when there is neither a *maximal sequent* nor applications of substitution rules in Π. Besides that, no major premise of Π is the conclusion of an application of $E_\vee$.

This restriction is due to the fact that an application of $E_\vee$ might "hide" a maximal sequent. For instance, take the derivation in figure 2.

$$\cfrac{\begin{array}{c}\Pi_1\\ \Gamma \vdash C \vee D\end{array} \quad \cfrac{\begin{array}{c}\Pi_2\\ \Gamma, C \vdash A\end{array} \quad \begin{array}{c}\Pi_3\\ \Gamma, C \vdash B\end{array}}{\Gamma, C \vdash A \wedge B}\ I_\wedge \quad \begin{array}{c}\Pi_4\\ \Gamma, D \vdash A \wedge B\end{array}}{\cfrac{\Gamma \vdash A \wedge B}{\Gamma \vdash A}\ E_\wedge}\ E_\vee$$

Fig. 2. Example of a derivation in ND

Such a derivation should not be considered as normal and it should be reduced first by commuting the elimination rule of the disjunction with that of the conjunction, which makes the maximal sequent $\Gamma, C \vdash A \wedge B$ evident.

2 Bijection

As we are, for now, restricting the work to normal and cut-free derivations, we are going to use the term "bijection" instead of "isomorphism". We say that ND and LJT are bijective when there exists transformations t_1 from ND to LJT and t_2 from LJT to ND, such that, if Π is a cut-free derivation in LJT, then $t_1(t_2(\Pi)) = \Pi$ and, if Π is a normal derivations in ND, then $t_2(t_1(\Pi)) = \Pi$. The *size* of a derivation Π (either in ND or in LJT) is the number of rules in Π.

The translations between ND and LJT are defined by induction on the size of derivations. One of the cases we need to take into account is when the last rule applied in a derivation is $\mathcal{D}$:

$$\cfrac{\begin{array}{c}\Sigma'\\ \Gamma, B; B \vdash C\end{array}}{\Gamma, B; \vdash C}\ \mathcal{D}$$

As was mentioned, the conclusions of the derivations in LJT have empty stoup, which means that Σ' is not a derivation. Hence, to define translations between derivations, we need to define translations between *pseudo-derivations*. Our translations are based on a pair (p, q) of functions where p is a map between pseudo-derivations and q is a map between derivations and when we define q, we may use p, and in some cases of the definition of p we use q.

Definition 2.1 [Left sequence] A sequence $\Gamma_1;\ B_1 \vdash A_1,\ \ldots,\ \Gamma_n;\ B_n \vdash A_n,\ \Gamma_{n+1};\ \Delta \vdash A_{n+1}$ of sequents of a cut-free derivation Π in LJT such that

- $\Gamma_{i-1}; B_{i-1} \vdash A_{i-1}, 1 < i \leq n$ is premiss of the left-rule of which $\Gamma_i; B_i \vdash A_i$ is the conclusion and
- $\Gamma_n; B_n \vdash A_n$ is premiss of the rule[7] of which $\Gamma_{n+1}; \Delta \vdash A_{n+1}$ is the conclusion

is called a *left sequence*. If $\Gamma_1; B_1 \vdash A_1$ is an initial sequent and $\Gamma_{n+1}; \vdash A_{n+1}$ is the conclusion of Π, then the sequence is called *major sequence* and Π is called a *pure left derivation*.

In other words, a pure left derivation is a derivation without occurrences of right-rules in its main branch. The derivation of figure 4 is a pure left derivation with the sequents $\Gamma; C \vdash C$, $\Gamma; B \to C \vdash C$, $\Gamma; A \to (B \to C) \vdash C$, $\Gamma; \vdash C$ forming its major sequence.

Definition 2.2 [Pseudo-derivation - LJT] Let Π be a pure left derivation in LJT and let $\Gamma_s; A_s \vdash B_s$ be a sequent of the major sequence of Π. A *pseudo-derivation* Σ of Π is the tree obtained from Π by removing every sequent occurrence below $\Gamma_s; A_s \vdash B_s$.

A pseudo-derivation can be seen as a pure left derivation where the bottom part is missing.

As an example,
$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\Gamma; A \vdash A}\,Ax}{\Gamma; \vdash A}\,\mathcal{D} \qquad \dfrac{}{\Gamma; B \vdash B}\,Ax}{\Gamma; A \to B \vdash B}\,{\to}{\vdash}}{\Gamma; \vdash B}\,\mathcal{D} \qquad \dfrac{}{\Gamma; C \vdash C}\,Ax}{\Gamma; B \to C \vdash C}\,{\to}{\vdash} \quad \text{and} \quad \dfrac{}{\Gamma; C \vdash C}\,Ax$$
are pseudo-derivations of the derivation of figure 4. As the derivation of figure 1 is not a pure left derivation, there is no pseudo-derivation associated to it.

Definition 2.3 [Elimination sequence] A sequence $\Gamma_1 \vdash A_1, \ldots, \Gamma_n \vdash A_n$ of a normal derivation Π in ND such that $\Gamma_i \vdash A_i, 1 \leq i < n$ is major premiss of the elimination rule of which $\Gamma_{i+1} \vdash A_{i+1}$ is the conclusion is called an *elimination sequence*. If $\Gamma_1 \vdash A_1$ is an initial sequent and $\Gamma_n \vdash A_n$ is the conclusion of Π, then the elimination sequence is called *major sequence* and Π is called a *pure elimination derivation*.

The derivation of figure 3 is a pure elimination derivation with the sequents $\Gamma \vdash A \to (B \to C)$, $\Gamma \vdash B \to C$, $\Gamma \vdash C$ forming its major sequence.

Definition 2.4 [Pseudo-derivation - ND] Let Π be a pure elimination derivation in ND and let $\Gamma \vdash B$ be a sequent in the major sequence of Π. A *pseudo-derivation*

[7] If $\Delta = \emptyset$, then it is a $\mathcal{D}$-rule; if $\Delta \neq \emptyset$, then it is a left rule.

Σ of Π is the tree obtained from Π by removing every sequent occurrence above $\Gamma \vdash B$.

A pseudo-derivation can be seen as an "unfinished" pure elimination derivation.

As an example, $\dfrac{\dfrac{\overline{\Gamma \vdash A}\,Ax \quad \overline{\Gamma \vdash A \to B}\,Ax}{\Gamma \vdash B}\,E_{\to} \quad \Gamma \vdash B \to C}{\Gamma \vdash C}\,E_{\to}$ and $\Gamma \vdash C$ are pseudo-derivations of the derivation of figure 3. As the derivation of figure 2 is not a pure elimination derivation, there is no pseudo-derivation associated to it.

Notation: we usually use Π to represent derivations and Σ to represent pseudo-derivations.

Lemma 2.5 *Let Π be a normal derivation in ND. If the bottommost rule applied in Π is an elimination rule, then Π is a pure elimination derivation.*

Proof. If Π were not a pure elimination derivation, then there would be a major premiss in Π that is also the conclusion of an introduction rule, that is, Π would not be normal. $\square$

From the previous lemma we infer that, as Π is a pure elimination derivation, its uppermost rule is also an elimination rule.

Now we define translations from LJT to ND. We are going to relate elimination rules with left rules. Note that the active formulas in the elimination rules are on the right side of an upper sequent while the active formulas in the left rules are on the left side of the bottom sequent (see figures 3 and 4). Additionally, left rules demand one formula more than elimination rules, that is, to define the elimination of implication, for instance, we used a set of formulas Γ and the formulas $\alpha \to \beta$, α and β. Besides these, in the left rule for implication we have a formula γ.

In a derivation, if we take the elimination sequence that contains a sequent $\Gamma \vdash \alpha \to \beta$, this formula γ appears in the last sequent, which will have the form $\Gamma' \vdash \gamma$. If we take the elimination sequence that contains the sequent $\Gamma; \alpha \to \beta \vdash \gamma$, this formula γ will appear in the right side of every sequent of this sequence.

As an example, compare the derivations of C from $\Gamma = \{A, A \to B, A \to (B \to C)\}$ in ND (figure 3) and in LJT (figure 4), where the bold formulas are the active formulas of the major sequence of the derivations.

$$\dfrac{\dfrac{\overline{\Gamma \vdash A}\,Ax \quad \overline{\Gamma \vdash A \to B}\,Ax}{\Gamma \vdash B}\,E_{\to} \quad \dfrac{\overline{\Gamma \vdash A}\,Ax \quad \overline{\Gamma \vdash \mathbf{A} \to (\mathbf{B} \to \mathbf{C})}\,Ax}{\Gamma \vdash \mathbf{B} \to \mathbf{C}}\,E_{\to}}{\Gamma \vdash \mathbf{C}}\,E_{\to}$$

Fig. 3. Example of a pure elimination derivation in ND

2.1 From LJT to ND

In this section, we define translations from LJT to ND.

Definition 2.6 Let Σ be a pseudo-derivation of a derivation Π in LJT. If g is a translation from cut-free derivations in LJT to normal derivations in ND, then the

$$
\dfrac{\dfrac{\dfrac{\dfrac{}{\Gamma; A \vdash A}\,Ax}{\Gamma; \vdash A}\,\mathcal{D} \qquad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\Gamma; A \vdash A}\,Ax}{\Gamma; \vdash A}\,\mathcal{D} \quad \dfrac{}{\Gamma; B \vdash B}\,Ax}{\Gamma; A \to B \vdash B}\,\to\vdash}{\Gamma; \vdash B}\,\mathcal{D} \quad \dfrac{}{\Gamma; \mathbf{C} \vdash C}\,Ax}{\Gamma; \mathbf{B} \to \mathbf{C} \vdash C}\,\to\vdash}{\Gamma; \mathbf{A} \to (\mathbf{B} \to \mathbf{C}) \vdash C}\,\to\vdash}{\Gamma; \vdash C}\,\mathcal{D}
$$

Fig. 4. Example of a pure elimination derivation in LJT

translation f of pseudo-derivations in LJT to pseudo-derivations in ND is defined recursively as follows:

If $\Sigma = \dfrac{}{\Gamma; C \vdash C}\,Ax$, then $f(\Sigma) = \Gamma \vdash C$.

If $\Sigma = \dfrac{\begin{array}{c}\Pi' \\ \Gamma; \vdash A\end{array} \quad \begin{array}{c}\Sigma' \\ \Gamma; B \vdash C\end{array}}{\Gamma; A \to B \vdash C}\,\to\vdash$, then $f(\Sigma) = \begin{array}{c}\dfrac{\begin{array}{c}g(\Pi') \\ \Gamma \vdash A\end{array} \quad \Gamma \vdash A \to B}{\Gamma \vdash B}\,E_{\to} \\ f(\Sigma')\end{array}$

If $\Sigma = \dfrac{\begin{array}{c}\Sigma' \\ \Gamma; B \vdash C\end{array}}{\Gamma; A \wedge B \vdash C}\,\wedge\vdash$, then $f(\Sigma) = \begin{array}{c}\dfrac{\Gamma \vdash A \wedge B}{\Gamma \vdash B}\,E_{\wedge} \\ f(\Sigma')\end{array}$

If $\Sigma = \dfrac{\begin{array}{c}\Sigma' \\ \Gamma; A \vdash C\end{array}}{\Gamma; A \wedge B \vdash C}\,\wedge\vdash$, then $f(\Sigma) = \begin{array}{c}\dfrac{\Gamma \vdash A \wedge B}{\Gamma \vdash A}\,E_{\wedge} \\ f(\Sigma')\end{array}$

If $\Sigma = \dfrac{\begin{array}{c}\Pi_1 \\ \Gamma, A; \vdash C\end{array} \quad \begin{array}{c}\Pi_2 \\ \Gamma, B; \vdash C\end{array}}{\Gamma; A \vee B \vdash C}\,\vee\vdash$,

then $f(\Sigma) = \dfrac{\Gamma \vdash A \vee B \quad \begin{array}{c}g(\Pi_1) \\ \Gamma, A \vdash C\end{array} \quad \begin{array}{c}g(\Pi_2) \\ \Gamma, B \vdash C\end{array}}{\Gamma \vdash C}\,E_{\vee}$

If $\Sigma = \dfrac{\begin{array}{c}\Sigma' \\ \Gamma; A \vdash C\end{array}}{\Gamma; \bot \vdash C}\,\bot\vdash$, then $f(\Sigma) = \begin{array}{c}\dfrac{\Gamma \vdash \bot}{\Gamma \vdash A}\,E_{\wedge} \\ f(\Sigma')\end{array}$

Note the role that the formula C plays on the translation. All the major sequents of the elimination sequence of Σ have conclusion C, but C "disappears" in the translation. It is so because the active formula of the major sequents in LJT are in the left side of the sequent while the active formulas in the major sequents of ND are on the right side of the sequent. The conclusion C only appears in the conclusion of the derivation which contains $f(\Sigma)$.

Lemma 2.7 *If g is a translation from cut-free derivations in LJT to normal derivations in ND, then, if Σ is a pseudo-derivation of a derivation in LJT, then $f(\Sigma)$ is a pseudo-derivation of a derivation in ND.*

Proof. The proof is by induction on the size of Σ and follows straight from the definition of f (definition 2.6). We show one case as an example:

Let $\Sigma = \dfrac{\begin{array}{cc}\Pi_1 & \Pi_2 \\ \Gamma, A; \vdash C & \Gamma, B; \vdash C\end{array}}{\Gamma; A \vee B \vdash C} \vee\vdash$. By definition 2.6, $f(\Sigma) =$

$$\dfrac{\Gamma \vdash A \vee B \quad \begin{array}{c} g(\Pi_1) \\ \Gamma, A \vdash C \end{array} \quad \begin{array}{c} g(\Pi_2) \\ \Gamma, B \vdash C \end{array}}{\Gamma \vdash C} E_{\vee} .$$

By hypothesis, both $g(\Pi_1)$ and $g(\Pi_2)$ are derivations in ND. Hence, $f(\Sigma)$ is a pseudo-derivation of a derivation in ND. □

Definition 2.8 Let Π be a cut-free derivation in LJT. The translation g from cut-free derivations in LJT to normal derivations in ND can be defined recursively as follows:

If $\Pi = \dfrac{\dfrac{}{\Gamma, A; A \vdash A} Ax}{\Gamma, A; \vdash A} \mathcal{D}$, then $g(\Pi) = \dfrac{}{\Gamma, A \vdash A} Ax$

If $\Pi = \dfrac{\begin{array}{ccc}\Pi_1 & & \Pi_n \\ \Gamma_1; \vdash B_1 & \dots & \Gamma_n; \vdash B_n\end{array}}{\Gamma; \vdash B} \vdash\odot$,

then $g(\Pi) = \dfrac{\begin{array}{ccc}g(\Pi_1) & & g(\Pi_n) \\ \Gamma_1 \vdash B_1 & \dots & \Gamma_n \vdash B_n\end{array}}{\Gamma \vdash B} I_{\odot}$, $\odot \in \{\to, \wedge, \vee\}$

If $\Pi = \dfrac{\begin{array}{c}\Sigma' \\ \Gamma, A; A \vdash B\end{array}}{\Gamma, A; \vdash B} \mathcal{D}$, then $g(\Pi) = \begin{array}{c}\dfrac{}{\Gamma, A \vdash A} Ax \\ f(\Sigma')\end{array}$

Theorem 2.9 *If Π is a cut-free derivation in LJT, then $g(\Pi)$ is a normal derivation in ND.*

Proof. The proof is by induction on the size of Σ and it follows straight from the definition 2.8 and lemma 2.7. We show one case as example:

Let $\Pi = \dfrac{\begin{array}{c}\Sigma' \\ \Gamma, A; A \vdash B\end{array}}{\Gamma, A; \vdash B} \mathcal{D}$. By definition 2.8, $g(\Pi) = \begin{array}{c}\dfrac{}{\Gamma, A \vdash A} Ax \\ f(\Sigma')\end{array}$.

As every sub-derivation of Σ' is smaller than Π, by induction hypothesis and by lemma 2.7, $f(\Sigma')$ is a pseudo-derivation of a derivation in ND. Hence, $g(\Pi)$ is a derivation in ND. □

2.2 From ND to LJT

In this section, we define translations from ND to LJT.

Definition 2.10 Let Σ be a pseudo-derivation of a pure elimination derivation Π in ND. If t is a translation from cut-free derivations in LJT to normal derivations in ND, then the translation s of pseudo-derivations in ND to pseudo-derivations in LJT is defined recursively as follows, where $\Gamma \vdash C$ is the conclusion of Π:

If $\Sigma = \Gamma \vdash C$, then $s(\Sigma) = \dfrac{}{\Gamma; C \vdash C}\,Ax$

If $\Sigma = \begin{array}{c}\dfrac{\Gamma \vdash A \wedge B}{\Gamma \vdash A}\,E_\wedge \\ \Sigma'\end{array}$, then $s(\Sigma) = \dfrac{\begin{array}{c}s(\Sigma')\\ \Gamma; A \vdash C\end{array}}{\Gamma; A \wedge B \vdash C}\,\wedge\vdash$

If $\Sigma = \begin{array}{c}\dfrac{\Gamma \vdash B \wedge A}{\Gamma \vdash A}\,E_\wedge \\ \Sigma'\end{array}$, then $s(\Sigma) = \dfrac{\begin{array}{c}s(\Sigma')\\ \Gamma; A \vdash C\end{array}}{\Gamma; B \wedge A \vdash C}\,\wedge\vdash$

If $\Sigma = \begin{array}{c}\dfrac{\begin{array}{c}\Pi'\\ \Gamma \vdash A\end{array} \quad \Gamma \vdash A \to B}{\Gamma \vdash B}\,E_\to \\ \Sigma'\end{array}$, then $s(\Sigma) = \dfrac{\begin{array}{c}t(\Pi')\\ \Gamma; \vdash A\end{array} \quad \begin{array}{c}s(\Sigma')\\ \Gamma; B \vdash C\end{array}}{\Gamma; A \to B \vdash C}\,\to\vdash$

If $\Sigma = \dfrac{\Gamma \vdash A \vee B \quad \begin{array}{c}\Pi_1\\ \Gamma, A \vdash D\end{array} \quad \begin{array}{c}\Pi_2\\ \Gamma, B \vdash D\end{array}}{\Gamma \vdash D}\,E_\vee$,

then $s(\Sigma) = \dfrac{\begin{array}{c}t(\Pi_1)\\ \Gamma, A; \vdash D\end{array} \quad \begin{array}{c}t(\Pi_2)\\ \Gamma, B; \vdash D\end{array}}{\Gamma; A \vee B \vdash D}\,\vee\vdash$

If $\Sigma = \begin{array}{c}\dfrac{\Gamma \vdash \bot}{\Gamma \vdash A}\,E_\bot \\ \Sigma'\end{array}$, then $t(\Sigma) = \dfrac{\begin{array}{c}s(\Sigma')\\ \Gamma; A \vdash C\end{array}}{\Gamma; \bot \vdash C}\,\wedge\vdash$

Lemma 2.11 *If t is a translation from normal derivations in ND to cut-free derivations of LJT, then, if Σ is a pseudo-derivation of ND, then $s(\Sigma)$ is a pseudo-derivation in LJT.*

Proof. The proof is by induction on the size of Σ and it follows straight from the definition of s (definition 2.10). We show one case as an example:

Let $\Sigma = \dfrac{\Gamma \vdash A \vee B \quad \begin{array}{c}\Pi_1\\ \Gamma, A \vdash D\end{array} \quad \begin{array}{c}\Pi_2\\ \Gamma, B \vdash D\end{array}}{\Gamma \vdash D}\,E_\vee$. By the definition 2.10, $s(\Sigma) =$

$$\dfrac{\begin{array}{c}t(\Pi_1)\\ \Gamma, A; \vdash D\end{array} \quad \begin{array}{c}t(\Pi_2)\\ \Gamma, B; \vdash D\end{array}}{\Gamma; A \vee B \vdash D}\,\vee\vdash$$

By hypothesis, both $t(\Pi_1)$ and $t(\Pi_2)$ are derivations in LJT. Hence, $s(\Sigma)$ is a pseudo-derivation of a derivation in ND. □

Definition 2.12 Let Π be a normal derivations in ND. The translation t from normal derivations in ND to cut-free derivations in LJT can be defined recursively as follows:

If $\Pi = \dfrac{}{\Gamma \vdash A}\,Ax$, then $t(\Pi) = \dfrac{\dfrac{}{\Gamma; A \vdash A}\,Ax}{\Gamma; \vdash A}\,\mathcal{D}$

$$\text{If } \Pi = \cfrac{\begin{array}{c}\Pi_1\\ \Gamma_1 \vdash C_1\end{array} \quad \dots \quad \begin{array}{c}\Pi_n\\ \Gamma_n \vdash C_n\end{array}}{\Gamma \vdash C}\, I_{\odot},$$

$$\text{then } t(\Pi) = \cfrac{\begin{array}{c}t(\Pi_1)\\ \Gamma_1; \vdash C_1\end{array} \quad \dots \quad \begin{array}{c}t(\Pi_n)\\ \Gamma_n; \vdash C_n\end{array}}{\Gamma; \vdash C}\, \odot\vdash, \ \odot \in \{\to, \wedge, \vee\}$$

The cases where the bottommost rule is an elimination rule can be easily derived from definition 2.10, but instead of looking to the bottommost rule, we refer to the lemma 2.5 and look at the uppermost rule applied in Π. We show one case as an example:

$$\text{If } \Pi = \begin{array}{c}\cfrac{\cfrac{}{\Gamma \vdash A \wedge B}\,Ax}{\Gamma \vdash A}\,E_{\wedge}\\ \Sigma'\end{array}, \text{ then } s(\Pi) = \cfrac{\cfrac{\begin{array}{c}s(\Sigma')\\ \Gamma; A \vdash C\end{array}}{\Gamma; A \wedge B \vdash C}\,\wedge\vdash}{\Gamma; \vdash C}\,\mathcal{D},$$

where $\Gamma \vdash C$ is the conclusion of Π and $A \wedge B \in \Gamma$.

Theorem 2.13 *If Π is a normal derivation in ND, then $t(\Pi)$ is a cut-free derivation in LJT.*

Proof. The proof is by induction on the size of Π and it follows straight from definition 2.12 and lemma 2.11. We show one case as an example:

$$\text{Let } \Pi = \begin{array}{c}\cfrac{\begin{array}{c}\Pi'\\ \Gamma \vdash A\end{array} \quad \cfrac{}{\Gamma \vdash A \to B}\,Ax}{\Gamma \vdash B}\,E_{\to}\\ \Sigma'\end{array}. \text{ By def. 2.12, } t(\Pi) = \cfrac{\cfrac{\begin{array}{c}t(\Pi')\\ \Gamma; \vdash A\end{array} \quad \begin{array}{c}s(\Sigma')\\ \Gamma; B \vdash C\end{array}}{\Gamma; A \to B \vdash C}\,\to\vdash}{\Gamma; \vdash C}\,\mathcal{D},$$

where $\Gamma \vdash C$ is the conclusion of Π and $A \to B \in \Gamma$.

As every sub-derivation of Σ' is smaller than Π, by lemma 2.11 $s(\Sigma')$ is a pseudo-derivation of a derivation in LJT and by induction hypothesis, $t(\Pi')$ is a cut-free derivation in LJT. Hence, $t(\Pi)$ is a derivation in LJT. □

2.3 The bijection

With the results from sections 2.1 and 2.2, we prove:

Lemma 2.14 *If $g(t(\Pi)) = \Pi$ for every normal derivation Π then, for every pseudo-derivation Σ of ND, $f(s(\Sigma)) = \Sigma$.*

Proof. The proof is by induction on the size of Σ and follows from the previous definitions and lemmas. We only show the case in which the last rule applied in Σ is $E_{\vee}$.

$$\text{If } \Sigma = \cfrac{\Gamma \vdash A \vee B \quad \begin{array}{c}\Pi_1\\ \Gamma, A \vdash C\end{array} \quad \begin{array}{c}\Pi_2\\ \Gamma, B \vdash C\end{array}}{\Gamma \vdash C}\,E_{\vee}, \text{ then } f\left(s\left(\Sigma\right)\right) =$$

$$f\left(\cfrac{\begin{array}{c}t(\Pi_1)\\ \Gamma, A; \vdash C\end{array} \quad \begin{array}{c}t(\Pi_2)\\ \Gamma, B; \vdash C\end{array}}{\Gamma; A \vee B \vdash C}\,\vee\vdash\right) = \cfrac{\Gamma \vdash A \vee B \quad \begin{array}{c}g(t(\Pi_1))\\ \Gamma, A \vdash C\end{array} \quad \begin{array}{c}g(t(\Pi_2))\\ \Gamma, B \vdash C\end{array}}{\Gamma \vdash C}\,E_{\vee}$$

By hypothesis, $g(t(\Pi_1)) = \Pi_1$ and $g(t(\Pi_2)) = \Pi_2$. Hence, $f(s(\Sigma)) = \Sigma$. □

Theorem 2.15 *For every normal derivation* Π *of ND,* $g(t(\Pi)) = \Pi$.

Proof. The proof is by induction on the size of Π and it follows from lemma 2.14. We only show the case in which the top-most rule applied in Π is $E_{\rightarrow}$.

$$\text{If } \Pi = \begin{array}{c} \dfrac{\overset{\Pi_1}{\Gamma \vdash A} \quad \dfrac{}{\Gamma \vdash A \rightarrow B}{\scriptstyle Ax}}{\Gamma \vdash B}{\scriptstyle E_{\rightarrow}} \\ \Sigma_1 \end{array}, \text{ then}$$

$$g\left(t\left(\Pi\right)\right) = g\left(\dfrac{\dfrac{\overset{t(\Pi_1)}{\Gamma;\vdash A} \quad \overset{s(\Sigma_1)}{\Gamma; B \vdash C}}{\Gamma; A \rightarrow B \vdash C}{\scriptstyle \rightarrow\vdash}}{\Gamma;\vdash C}{\scriptstyle \mathcal{D}} \right) = \begin{array}{c} \dfrac{\overset{g(t(\Pi_1))}{\Gamma \vdash A} \quad \dfrac{}{\Gamma \vdash A \rightarrow B}{\scriptstyle Ax}}{\Gamma \vdash B}{\scriptstyle E_{\rightarrow}} \\ f(s(\Sigma_1)) \end{array}$$

By IH, $g(t(\Pi_1)) = \Pi_1$ and by lemma 2.14 and IH, $f(s(\Sigma_1)) = \Sigma_1$. Hence, $g(t(\Pi)) = \Pi$. □

Lemma 2.16 *If* $t(g(\Pi)) = \Pi$ *for every cut-free derivation* Π *then, for every pseudo-derivation* Σ *of LJT,* $s(f(\Sigma)) = \Sigma$

Proof. The proof is by induction on the size of Σ and it follows from the previous definitions and lemmas. We only show the case in which the last rule applied in Σ is $\mathcal{D}$ preceded by $\vee\vdash$.

$$\text{If } \Sigma = \dfrac{\overset{\Pi_1}{\Gamma, A;\vdash C} \quad \overset{\Pi_2}{\Gamma, B;\vdash C}}{\Gamma; A \vee B \vdash C}{\scriptstyle \vee\vdash}, \text{ then } s\left(f\left(\Sigma\right)\right) =$$

$$s\left(\dfrac{\Gamma \vdash A \vee B \quad \overset{g(\Pi_1)}{\Gamma, A \vdash C} \quad \overset{g(\Pi_2)}{\Gamma, B \vdash C}}{\Gamma \vdash C}{\scriptstyle \vee\vdash} \right) = \dfrac{\overset{t(g(\Pi_1))}{\Gamma, A;\vdash C} \quad \overset{t(g(\Pi_2))}{\Gamma, B;\vdash C}}{\Gamma;\vdash A \vee B}{\scriptstyle \vee\vdash}$$

By hypothesis, $t(g(\Pi_1)) = \Pi_1$ and $t(g(\Pi_2)) = \Pi_2$. Hence, $s(f(\Sigma)) = \Sigma$. □

Now, we proceed to prove that the translations are inverse to each other.

Theorem 2.17 *For every cut-free derivation* Π *of LJT,* $t(g(\Pi)) = \Pi$.

Proof. The proof is by induction on the size of Π and it follows from lemma 2.14. We only show the case in which the last rule applied in Π is $\mathcal{D}$ preceded by $\rightarrow\vdash$.

$$\text{If } \Pi \text{ is of the form } \dfrac{\dfrac{\overset{\Pi'}{\Gamma;\vdash A} \quad \overset{\Sigma'}{\Gamma; B \vdash C}}{\Gamma; A \rightarrow B \vdash C}{\scriptstyle \rightarrow\vdash}}{\Gamma;\vdash C}{\scriptstyle \mathcal{D}}, \text{ then } t\left(g\left(\Pi\right)\right)$$

$$= t\left(\begin{array}{c} \dfrac{\overset{g(\Pi')}{\Gamma \vdash A} \quad \dfrac{}{\Gamma \vdash A \rightarrow B}{\scriptstyle Ax}}{\Gamma \vdash B}{\scriptstyle E_{\rightarrow}} \\ f(\Sigma') \end{array} \right) = \dfrac{\dfrac{\overset{t(g(\Pi'))}{\Gamma;\vdash A} \quad \overset{s(f(\Sigma'))}{\Gamma; B \vdash C}}{\Gamma; A \rightarrow B \vdash C}{\scriptstyle \rightarrow\vdash}}{\Gamma;\vdash C}{\scriptstyle \mathcal{D}}$$

By induction hypothesis, $t(g(\Pi')) = \Pi'$ and by lemma 2.16, $s(f(\Sigma')) = \Sigma'$. □

From theorems 2.15 and 2.17, we have that the translations defined between ND and LJT are bijective.

3 Considering non-normal derivations

In this section we extend the mappings shown in the previous section to deal with non-normal ND derivations and LJT with cuts. Due to lack of space, the presentation omits the proofs. We hope its reading is enough to make the reader understand how the isomorphism we obtained between LJT and ND works.

First, we extend the definitions by allowing the sequents in left/elimination sequences to be premises of cut/substitution rules. This automatically changes the definition of pseudo-derivations allowing them to have occurrences of cut/substitution rules.

Second, we define conversion steps for both pseudo-derivations and derivations and show the normalization property for both LJT and ND. The conversions will either eliminate the cut/substitution rule, bring it one level up in the derivation or decrease the number of connectives in the cut-formula. Some examples of conversions are:

In LJT:

$$\dfrac{\begin{matrix}\Pi_1\\ \Gamma;\vdash A\end{matrix} \quad \dfrac{\begin{matrix}\Pi_{21}\\ \Gamma,A;\vdash P\end{matrix} \quad \begin{matrix}\Pi_{22}\\ \Gamma,A;\vdash Q\end{matrix}}{\Gamma,A;\vdash P\wedge Q}\vdash\wedge}{\begin{matrix}\Gamma;\vdash P\wedge Q\\ \Psi_3\end{matrix}}C_M \;\rhd\; \dfrac{\dfrac{\begin{matrix}\Pi_1\\ \Gamma;\vdash A\end{matrix} \quad \begin{matrix}\Pi_{21}\\ \Gamma,A;\vdash P\end{matrix}}{\Gamma;\vdash P}C_M \quad \dfrac{\begin{matrix}\Pi_1\\ \Gamma;\vdash A\end{matrix} \quad \begin{matrix}\Pi_{22}\\ \Gamma,A;\vdash Q\end{matrix}}{\Gamma;\vdash Q}C_M}{\begin{matrix}\Gamma;\vdash P\wedge Q\\ \Psi_3\end{matrix}}\vdash\wedge$$

$$\dfrac{\begin{matrix}\Pi_1\\ \Gamma;\vdash A\end{matrix} \quad \dfrac{\begin{matrix}\Pi_{21}\\ \Gamma,A;\vdash P\end{matrix} \quad \begin{matrix}\Pi_{22}\\ \Gamma,A;Q\vdash B\end{matrix}}{\Gamma,A;P\to Q\vdash B}\to\vdash}{\begin{matrix}\Gamma;P\to Q\vdash B\\ \Psi_3\end{matrix}}C_M \;\rhd\; \dfrac{\dfrac{\begin{matrix}\Pi_1\\ \Gamma;\vdash A\end{matrix} \quad \begin{matrix}\Pi_{21}\\ \Gamma,A;\vdash P\end{matrix}}{\Gamma;\vdash P}C_M \quad \dfrac{\begin{matrix}\Pi_1\\ \Gamma;\vdash A\end{matrix} \quad \begin{matrix}\Pi_{22}\\ \Gamma,A;Q\vdash B\end{matrix}}{\Gamma;Q\vdash B}C_M}{\begin{matrix}\Gamma;P\to Q\vdash B\\ \Psi_3\end{matrix}}\to\vdash$$

In ND:

$$\dfrac{\dfrac{}{\Gamma,A\vdash A}Ax \quad \begin{matrix}\Psi_2\\ \Gamma,A\vdash B\end{matrix}}{\begin{matrix}\Gamma,A\vdash B\\ \Psi_3\end{matrix}}S \;\rhd\; \begin{matrix}\Psi_2\\ \Gamma,A\vdash B\\ \Psi_3\end{matrix}$$

$$\dfrac{\begin{matrix}\Psi_1\\ \Gamma\vdash A\end{matrix} \quad \dfrac{\begin{matrix}\Psi_{21}\\ \Gamma,A\vdash P\end{matrix} \quad \begin{matrix}\Psi_{22}\\ \Gamma,A\vdash Q\end{matrix}}{\Gamma,A\vdash P\wedge Q}I_\wedge}{\begin{matrix}\Gamma\vdash P\wedge Q\\ \Psi_3\end{matrix}}S \;\rhd\; \dfrac{\dfrac{\begin{matrix}\Psi_1\\ \Gamma\vdash A\end{matrix} \quad \begin{matrix}\Psi_{21}\\ \Gamma,A\vdash P\end{matrix}}{\Gamma\vdash P}S \quad \dfrac{\begin{matrix}\Psi_1\\ \Gamma\vdash A\end{matrix} \quad \begin{matrix}\Psi_{22}\\ \Gamma,A\vdash Q\end{matrix}}{\Gamma\vdash Q}S}{\begin{matrix}\Gamma\vdash P\wedge Q\\ \Psi_3\end{matrix}}I_\wedge$$

The proof of the normalization is by induction on the complexity of the cut/substitution formula and by the size of the (pseudo-)derivation and it uses similar lemmas regarding pseudo-derivations.. Thus, the following theorems hold:

Theorem 3.1 *Every derivation Π in LJT can be transformed into a cut-free derivation.*

Theorem 3.2 *Every derivation in ND can be transformed into a normal derivation.*

We define maps f' and s' between pseudo-derivations and g' and t' between derivations and show that these maps are indeed translations. When the last rule applied in Σ/Π is not a cut/substitution rule, then the definition of f', g', s' and t' are similar to those of f, g, s and t, the only difference being that Σ and Π need not be cut-free nor normal.

We give some examples for when the last rule applied in Σ/Π is a cut/substitution rule:

Definition 3.3 Let Σ be a pseudo-derivation of a derivation Π in LJT. If g' is a translation from derivations in LJT to derivations in ND then the translation f' from pseudo-derivations of derivations in LJT to pseudo-derivations of derivations in ND is defined recursively as follows:

$$\text{(i)}\ \ f'\left(\dfrac{\dfrac{\begin{matrix}\Sigma_1'\\ \Gamma;Q\vdash A\end{matrix}}{\Gamma;P\wedge Q\vdash A}\,r_1 \quad \dfrac{\Sigma_2}{\Gamma;A\vdash B}\,r_2}{\Gamma;P\wedge Q\vdash B}\,C_H\right) = f'\left(\dfrac{\dfrac{\begin{matrix}\Sigma_1'\\ \Gamma;Q\vdash A\end{matrix} \quad \dfrac{\Sigma_2}{\Gamma;A\vdash B}\,r_2}{\Gamma;Q\vdash B}\,C_H}{\Gamma;P\wedge Q\vdash B}\,r_1\right)$$

$$\text{(ii)}\ \ f'\left(\dfrac{\begin{matrix}\Pi_1\\ \Gamma;\vdash A\end{matrix} \quad \dfrac{\Sigma_2}{\Gamma,A;C\vdash B}\,\odot\vdash}{\Gamma;C\vdash B}\,C_M\right) = \dfrac{\dfrac{\begin{matrix}g'(\Pi_1)\\ \Gamma\vdash A\end{matrix} \quad \Gamma,A\vdash C}{\Gamma\vdash C}\,S}{f'(\Sigma_2)}\,E_\odot,\ \odot\in\{\wedge,\vee,\rightarrow,\bot\}$$

Definition 3.4 Let Π be a derivation in LJT. The transformation g' from derivations in LJT to derivations in ND is defined recursively as follows:

$$\text{(i)}\ \ g'\left(\dfrac{\dfrac{\Pi_1}{\Gamma;\vdash C}\,\vdash\odot \quad \dfrac{\Sigma_2}{\Gamma;C\vdash B}\,\odot\vdash}{\Gamma;\vdash B}\,C_H\right) = \dfrac{\dfrac{g'(\Pi_1)}{\Gamma\vdash C}\,I_\odot}{f'(\Sigma_2)}\,E_\odot,\ \odot\in\{\wedge,\vee,\rightarrow,\bot\}$$

$$\text{(ii)}\ \ g'\left(\dfrac{\dfrac{\Pi_1}{\Gamma;\vdash A}\,r_1 \quad \dfrac{\begin{matrix}\Sigma_2'\\ \Gamma,A;C\vdash B\end{matrix}}{\Gamma,A;\vdash B}\,\mathcal{D}}{\Gamma;\vdash B}\,C_M\right) = g'\left(\dfrac{\dfrac{\dfrac{\Pi_1}{\Gamma;\vdash A}\,r_1 \quad \begin{matrix}\Sigma_2'\\ \Gamma,A;C\vdash B\end{matrix}}{\Gamma;C\vdash B}\,C_M}{\Gamma;\vdash B}\,\mathcal{D}\right)$$

Definition 3.5 Let Σ be a pseudo-derivation of a derivation Π in ND. If t' is a translation from derivations in ND to derivations in LJT then the translation s' from pseudo-derivations of derivations in ND to pseudo-derivations of derivation in LJT is defined recursively as follows:

$$s'\left(\begin{matrix}\dfrac{\begin{matrix}\Pi_1\\ \Gamma\vdash A\end{matrix} \quad \Gamma,A\vdash C}{\Gamma\vdash C}\,S\\ \Sigma_3\end{matrix}\right) = \dfrac{\begin{matrix}t'(\Pi_1)\\ \Gamma;\vdash A\end{matrix} \quad \begin{matrix}s'(\Sigma_3^A)\\ \Gamma,A;C\vdash B\end{matrix}}{\Gamma;C\vdash B}\,C_M,$$

where $\Gamma\vdash B$ is the conclusion of the derivation to whom s' is applied. Remember that, from the definition of pseudo-derivation, both $\Gamma, A\vdash C$ and $\Gamma\vdash C$ are major premises.

Definition 3.6 Let Π be a derivation in ND. The transformation t' from derivations in LJT to derivations in ND is defined recursively as follows:

$$\text{(i)}\ \ \Pi = \dfrac{\dfrac{\Pi_1}{\Gamma\vdash C}\,I_\odot}{\Sigma_2}\,E_\odot \text{ and } t'(\Pi) = \dfrac{\dfrac{t'(\Pi_1)}{\Gamma;\vdash C}\,\vdash\odot \quad \dfrac{s'(\Sigma_2)}{\Gamma;C\vdash E}\,\odot\vdash}{\Gamma;\vdash E}\,C_H.$$

(ii) $$t'\left(\dfrac{\begin{array}{cc}\Pi_1 & \Pi_2 \\ \Gamma \vdash A & \Gamma, A \vdash B\end{array}}{\Gamma \vdash B}\,\mathcal{S}\right) = \dfrac{\begin{array}{cc}t'(\Pi_1) & t'(\Pi_2) \\ \Gamma; \vdash A & \Gamma, A; \vdash B\end{array}}{\Gamma; \vdash B}\,C_M$$

From these definitions, we prove results by induction on the length of Σ/Π and on the rank of the cut-formulas of Σ/Π (if any). The rank of a rule r in a (pseudo-)derivation Σ/Π is the number of rules applied in Σ/Π above r up to the top-formula. We only show the proof for the cases shown in the definitions above.

Lemma 3.7 *If g' is a translation from derivations in LJT to derivations in ND, then, if Σ is a pseudo-derivation of a derivation in LJT, then $f'(\Sigma)$ is a pseudo-derivation of a derivation in ND.*

Proof. Given the two items shown in definition 3.3, we have that:

(i) the rank of the last occurrence of C_H is smaller on the derivation on the right side of the equality sign. Hence, by induction hypothesis, it can be translated into a derivation in ND.

(ii) from the hypothesis, $g'(\Pi_1)$ is a derivation in ND and by the induction hypothesis, $f'(\Sigma_2)$ is a pseudo-derivation of a derivation in ND. Hence, the derivation on the right side of the equality sign is a derivation in ND.

□

Theorem 3.8 *If Π is a derivation in LJT, then $g'(\Pi)$ is a derivation in ND.*

Proof. Given the two items shown in definition 3.4, we have that:

(i) by induction hypothesis, $g'(\Pi_1)$ is a derivation in ND and by lemma 3.7, $f'(\Sigma_2)$ is a pseudo-derivation of a derivation in ND. Hence, the derivation on the right side of the equality sign is a derivation in ND.

(ii) the rank of the last occurrence of C_M is smaller on the derivation on the right side of the equality sign. Hence, by induction hypothesis, it can be translated into a derivation in ND.

□

Lemma 3.9 *If t' is a translation from derivations in ND to derivations in LJT, then, if Σ is a pseudo-derivation of a derivation in ND, then $s'(\Sigma)$ is a pseudo-derivation of a derivation in LJT.*

Proof. Given the example shown in definition 3.5, we have that, by hypothesis, $t'(\Pi_1)$ is a derivation in LJT and, by induction hypothesis, $s'(\Sigma_3^A)$ [8] is a pseudo-derivation of a derivation in LJT. Hence, the pseudo-derivation is a pseudo-derivation of a derivation in LJT. □

Theorem 3.10 *If Π is a derivation in ND, then $t'(\Pi)$ is a derivation in LJT.*

Proof. Given the two items shown in definition 3.6, we have that:

[8] Σ_3^A is the result of adding the formula A in the premises of all sequents in Σ_3.

(i) By induction hypothesis, $t'(\Pi_1)$ is a derivation in LJT and by lemma 3.9, $s'(\Sigma_2)$ is a pseudo-derivation of a derivation in LJT. Hence, $t'(\Pi')$ is a derivation in LJT.

(ii) By induction hypothesis, both $t'(\Pi_1)$ and $t'(\Pi_2)$ are derivations in LJT. Hence, the derivations of item (ii) are derivations in LJT.

□

To define an isomorphism between ND and LJT, we use the following notion of equivalence:

Definition 3.11 We say that two (pseudo-)derivations Π and Π' are *equivalent* if Π and Π' reduce to a same normal/cut-free (pseudo-)derivation.

By induction on the size of derivations and using similar lemmas regarding pseudo-derivations, we can prove the following results:

Theorem 3.12 *For every derivation* Π *in LJT,* $t'(g'(\Pi)) \approx \Pi$.

Theorem 3.13 *For every derivation* Π *in ND,* $g'(t'(\Pi)) \approx \Pi$.

From these results we conclude that the translations g' and t' form an isomorphism (modulo $\approx$) between LJT and ND.

4 Conclusion

We achieved a bijection between normal and cut-free derivations. In order to complete the proof-theoretical isomorphism between Natural Deduction and Sequent Calculus, the translations (f, g) and (s, t) were extended to translate any derivation, and not just normal and cut-free ones. Finally, we showed that the extended translations are bijective.

As future work, we suggest the development of the bijection by showing translation between the λ-calculus and a term notation for LJT (as, for example, Herbelin's syntax in [5]).

References

[1] Vincent Danos, Jean-Baptiste Joinet, and Harold Schellinx. Lkq and lkt: Sequent calculi for second order logic based upon dual linear decompositions of classical implication.

[2] G. Gentzen. Investigations into logical deduction. In M. E. Szabo, editor, *The collected papers of Gerhard Gentzen*. North Holland, 1969.

[3] J.Y. Girard, P. Taylor, and Y. Lafont. *Proofs and Types*. Cambridge Tracts in Theoretical Computer Science 7. Cambridge University Press, 1989.

[4] Anders Starcke Henriksen. Using ljf as a framework for proof systems. Technical report, 2010.

[5] Hugo Herbelin. A lambda-calculus structure isomorphic to gentzen-style sequent calculus structure. In *Selected Papers from the 8th International Workshop on Computer Science Logic*, CSL '94, pages 61–75, London, UK, UK, 1995. Springer-Verlag.

[6] Hugo Herbelin. *Séquents qu'on calcule: de l'interprétation du calcul des séquents comme calcul de λ-termes et comme calcul de stratégies gagnantes*. Ph.D. thesis, University Paris 7, January 1995.

[7] J.-B. Joinet. *Étude de la Normalisation du Calcul des Séquents Classique à Travers la Logique Linéaire.* PhD thesis, University of Paris VII, 1993.

[8] Dale Miller and Alexis Saurin. From proofs to focused proofs: A modular proof of focalization in linear logic. In *CSL*, pages 405–419, 2007.

[9] S. Negri, J. von Plato, and A. Ranta. *Structural Proof Theory.* Cambridge University Press, 2008.

[10] Vivek Nigam and Dale Miller. A framework for proof systems. *J. Autom. Reasoning*, 45(2):157–188, 2010.

[11] Garrel Pottinger. Normalization as a homomorphic image of cut-elimination. *Annals of Mathematical Logic*, 12:323–357, 1977.

[12] D. Prawitz. *Natural deduction: a proof-theoretical study.* PhD thesis, 1965.

[13] J. Zucker. The correspondence between cut-elimination and normalization I. 7(1):1–112, 1974.