Superdeduction as a logical framework

Guillaume Burel

Nancy-Université (Université Henri Poincaré) & LORIA (UMR 7503 CNRS-INPL-INRIA-Nancy2-UHP)
E-mail: `guillaume.burel@ens-lyon.org`

Abstract

Superdeduction is a formalism closely related to deduction modulo which permits to enrich a first-order deduction system—such as natural deduction or sequent calculus—with new inference rules automatically computed from the presentation of a theory. We give a natural encoding from every Pure Type System (PTS) into superdeduction by defining an appropriate first-order theory. We prove that this translation is correct and conservative, showing a correspondence between valid typing judgment in the PTS and provable sequents in the corresponding superdeductive system. As a byproduct, we also introduce the superdeductive sequent calculus for intuitionistic *logic, which was until now only defined for classical logic. We show its equivalence with the superdeductive natural deduction. This implies that superdeduction can be easily used as a logical framework. These results lead to a better understanding of the implementation and the automation of proof search for PTS, as well as to more cooperation between proof assistants.*

1. Introduction

According to Pfenning [24], a logical framework is a meta-language for the specification of deductive systems. The most famous one is the Edinburgh Logical Framework introduced in [21], which is based on a λ-calculus with dependent types which is known as λP or $\lambda\Pi$. Many systems, such as first-order natural deduction, simply typed λ-calculus, sundry presentations of modal logics, etc. can be represented in it.

Most existing interactive theorem provers are based on the logical framework approach, either explicitly as in Isabelle (`http://isabelle.in.tum.de/`) or implicitly in proof assistants based on extensions of $\lambda\Pi$, such as Coq (`http://coq.inria.fr/`). These extensions are instances of what are called Pure Type Systems (PTS), which are generic typing systems for λ-calculi with dependent types.

Superdeduction [27, 8] consists quite in the inverse ap- proach to logical frameworks: from the presentation of a theory is computed a deductive system which is well adapted to proof representation and search in this theory. For instance, from the intuitive definition of the inclusion $A \subseteq B \equiv (\forall x.\; x \in A \Rightarrow x \in B)$ are computed the two rules

$$\subseteq_I^{\mathsf{df}} \frac{\Gamma, x \in A \vdash x \in B}{\Gamma \vdash A \subseteq B} \;_{\substack{x \text{ not} \\ \text{free in } \Gamma}} \qquad \subseteq_E^{\mathsf{df}} \frac{\Gamma \vdash A \subseteq B \quad \Gamma \vdash t \in A}{\Gamma \vdash t \in B}$$

in supernatural deduction, the superdeductive system based on natural deduction.

Although superdeduction originally comes from typed ρ-calculus [10], it shares many aspects with deduction modulo [15]. Deduction modulo was at first thought of as a way to make computation and deduction easily cooperate in proofs. This is done by modeling the computations through a congruence between propositions, and by applying the inference rules of the deductive system we are considering modulo this congruence. For the instance of the inclusion above, this means that the following derivation is valid in natural deduction modulo:

$$\forall_I \frac{\Gamma \vdash x \in A \Rightarrow x \in B}{\Gamma \vdash A \subseteq B} \;_{\substack{x \text{ not free in } \Gamma \\ A \subseteq B \equiv (\forall x.\; x \in A \Rightarrow x \in B)}}$$

We can remark that above this derivation it is quite natural to apply $\Rightarrow_I$. This explains why the new rule $\subseteq_I^{\mathsf{df}}$, where this step is precomputed, is more adapted to build proofs in the theory.

Notwithstanding, superdeduction and deduction modulo can be seen as a logical framework. In particular, HOL was presented in deduction modulo [14], and this is also possible in superdeduction, leading to new inference rules which correspond exactly to the ones of HOL-λ. This presentation is not only interesting for practical reasons, leading to automated theorem proving procedures based on deduction modulo, such as ENAR [15] or TaMed [6], and well adapted for HOL. It also permits to get a first-order characterization of HOL. Indeed, proving in the sequent calculus modulo, or in supernatural deduction, is equivalent to proving in the pure sequent calculus within a so called compatible (first-order) theory (see [15, Proposition 1.8]).

An important issue in interactive theorem proving is the abundance of proof assistants. Ideally, we would like to

be able to use theories developed in a particular theorem prover into another one, and thus to have translations between all the proof systems corresponding to the provers. Another approach was proposed by Kirchner through Fellowship [25]. This tool permits to translate proofs in the first-order sequent calculus into proofs in PVS (http://pws.csl.sri.com/), Coq (http://coq.inria.fr/) and soon other provers. Recently, another step was performed by Cousineau and Dowek [11] who proved that every Pure Type System can be encoded in $\lambda\Pi$ modulo. Inversely to Fellowship, this permits to translate proofs of the theorem provers into proofs of $\lambda\Pi$ modulo, and the tool Europa, which checks proofs in $\lambda\Pi$ modulo, can then be used. The question that naturally arises is why this encoding is not performed in a first-order system (i.e. without β-conversion, which is a higher-order mechanism) such as for instance the sequent calculus modulo, as was done for HOL.

This paper shows how it is possible. Then, using the equivalence between proving in deduction modulo (or in superdeduction) and proving using a compatible theory, this will give us proofs in the first-order sequent calculus (without modulo), which can be fed to Fellowship. Furthermore, this leads to the automation of proof search in PTS, either through automated provers based on deduction modulo (TaMed and ENAR), or, using the equivalence, through standard automated provers.

To encode the PTS, we define a first-order theory which produces, in supernatural deduction, new inference rules that are close to the typing rules of the PTS. In particular, in a PTS, there is a conversion rule that says that two β-convertible terms type the same terms. To simulate this in a first-order setting, we will use some λ-calculus with explicit substitutions, such as for instance λ_σ [1]. As we want our work to be independent of a particular calculus, we use a scheme of calculi, quite the same as Kesner [22] (who used it to get a generic proof of confluence of such calculi). We then prove that our translation is correct—valid typing judgments of the PTS can be translated into provable sequents in supernatural deduction—and conservative—provable sequents in supernatural deduction corresponds (modulo β-reduction) to valid typing judgments.

To prove the conservativeness, we need the fact that the newly created rules are enough to prove everything related to the first-order theory. To prove this, we define correspondences between supernatural deduction and the superdeductive system for the intuitionistic sequent calculus. Being until now only introduced for classical logic [8], we need to define the so-called intuitionistic super sequent calculus here. Because we need as few permutation problems as possible, we do not base it on Gentzen sequent calculus LJ, but on the sequent calculus LB [26]. In addition to a better comprehension of the properties of the supernatural deduction as a logical framework, this also leads to the actual use of
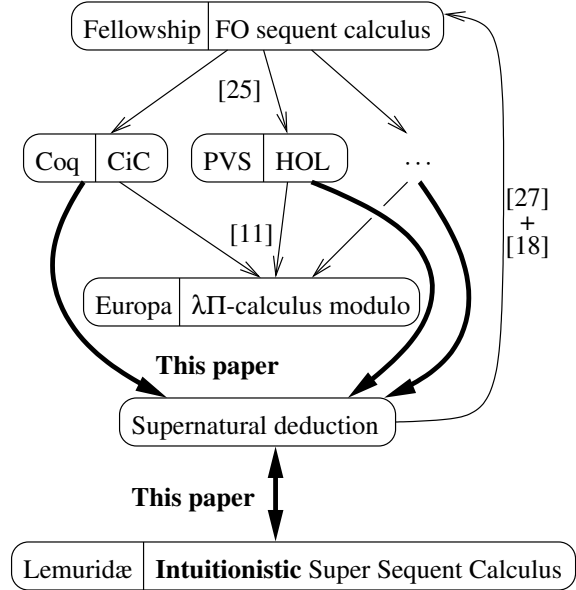
Figure 1. Complementarity of our work with existing systems

its super sequent calculus counterpart in the theorem prover based on superdeduction, Lemuridæ [8], in a proven sound and correct way.

The contributions of this paper are summed up in figure 1. We show that every PTS can be encoded in a natural way in supernatural deduction through the definition of a first-order theory. We prove that this translation is correct and conservative. This shows that superdeduction is expressive enough to encode elaborated deductive systems, and can therefore be considered as a powerful logical framework. We also prove an equivalence between supernatural deduction and super sequent calculus, and this implies that the newly computed rules are enough to prove results related to the theory, i.e. we do not need logical rules.

The next section presents the notions necessary to the remaining of the paper, i.e. LB, PTS, and the scheme of calculi with explicit substitutions. Section 3 introduces the two systems of superdeduction, i.e. supernatural deduction and super sequent calculus. The translation between is given, as well as a theorem (Corollary 7) stating that the new inference rules are enough to prove results related only to the theory used to compute the rules, for a large class of theories. Section 4 defines the first-order theory used to encode a PTS, and presents the supernatural deduction rules computed from it. Their soundness and conservativeness is then proved. Finally, Section 5 presents some results following from this encoding. The appendix contains postponed proofs.

2. Prerequisites

The reader is assumed to be familiar with the natural deduction NJ and the sequent calculi LJ and LK of Gentzen [18], and to have basic notions on rewriting, as can be found in [3], which notations we will use. We restrict ourselves to intuitionistic first-order logic with only $\Rightarrow, \wedge$ as connectors and $\forall$ as quantifier. This is enough for the remaining of the paper, but this could be generalized to full intuitionistic first-order logic. $\{A/x\}B$ denotes the result of substituting all occurrences of x by A in B.

2.1. Intuitionistic sequent calculus

The standard sequent calculus for first-order intuitionistic logic is the one introduced by Gentzen [18]. It is the sequent calculus for first-order classical logic restricted to sequents with at most one formula in the right part. Unfortunately, this calculus permits too few permutations between inference rules, and therefore the super sequent calculus based on it would be incomplete even for simple theories. We will therefore use another sequent calculus, LB [26], which is based on a tableau method for intuitionistic logic and allows more permutations. The inference rules for LB are the following:

$$\text{Axiom } \frac{}{\Gamma, A \vdash A, \Delta} \qquad \text{Cut } \frac{\Gamma, A \vdash \Delta \qquad \Gamma \vdash A, \Delta}{\Gamma \vdash \Delta}$$

$$\wedge_L \frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta} \qquad \wedge_R \frac{\Gamma \vdash A, \Delta \qquad \Gamma \vdash B, \Delta}{\Gamma \vdash A \wedge B, \Delta}$$

$$\Rightarrow_L \frac{\Gamma, B \vdash \Delta \qquad \Gamma \vdash A, \Delta}{\Gamma, A \Rightarrow B \vdash \Delta} \qquad \Rightarrow_R \frac{\Gamma, A \vdash B}{\Gamma \vdash A \Rightarrow B, \Delta}$$

$$\forall_L \frac{\Gamma, \{t/x\}A \vdash \Delta}{\Gamma, \forall x.\, A \vdash \Delta} \qquad \forall_R \frac{\Gamma \vdash A}{\Gamma \vdash \forall x.\, A, \Delta} \text{ } x \text{ not free in } \Gamma$$

Note how, on $\Rightarrow_R$ and $\forall_R$, Δ is discarded in the above sequent. This is why this sequent calculus corresponds to intuitionistic logic. Axiom and Cut are called *identity rules*, the others *logical rules*.

2.2. Pure Type Systems

PTS are typing systems for the λ-calculus with dependent type. In particular, the simply-typed λ-calculus, $\lambda\Pi$, and the calculus of constructions (in fact, the whole λ-cube), λHOL, etc. can be considered as PTS. We refer to [4].

Definition 1 (Pure Type Systems). *A PTS is defined by three data: a set $\mathcal{S}$ of* sorts*, a relation $\mathcal{A} \subseteq \mathcal{S} \times \mathcal{S}$ of* axioms*, and a relation $\mathcal{R} \subseteq \mathcal{S} \times \mathcal{S} \times \mathcal{S}$ of* rules*.*
Pseudo-terms are defined using this grammar ($s \in \mathcal{S}$):

$$T \overset{\text{def}}{=} x \mid s \mid \Pi x : T\, T \mid \lambda x : T\, T \mid (T\, T) \ .$$

The typing rules are represented in figure 2. A term of the PTS is a pseudo-term which can be typed.

Any PTS has the following properties:

Lemma 1. Subject reduction *[4, Theorem 5.2.15]*
If $\Gamma \vdash_{\text{PTS}} A : B$ and $A \overset{*}{\longrightarrow}_{\beta} A'$ then $\Gamma \vdash_{\text{PTS}} A' : B$.

Subject reduction for η *[19, Proposition 2.6]*
If $\Gamma \vdash_{\text{PTS}} \lambda x : A\, (T\, x) : B$ then $\Gamma \vdash_{\text{PTS}} T : B$.

Correctness of types *[4, Corollary 5.2.14] If $\Gamma \vdash_{\text{PTS}} A : B$ then there exist some $s \in \mathcal{S}$ such that either $B = s$ or $\Gamma \vdash_{\text{PTS}} B : s$.*

Substitution *[4, Lemma 5.2.11] If $\Gamma[x : B] \vdash_{\text{PTS}} \alpha$ and $\Gamma \vdash_{\text{PTS}} A : B$ then $\Gamma \vdash_{\text{PTS}} \{A/x\}\alpha$.*

Definition 2. *A PTS is* functional *if ($\langle s_1, s \rangle \in \mathcal{A}$ and $\langle s_1, s' \rangle \in \mathcal{A}$) or ($\langle s_1, s_2, s \rangle \in \mathcal{R}$ and $\langle s_1, s_2, s' \rangle \in \mathcal{R}$) implies $s = s'$.*

Lemma 2 (Uniqueness [4, Lemma 5.2.21]). *If a PTS is functional, $\Gamma \vdash_{\text{PTS}} A : B$ and $\Gamma \vdash_{\text{PTS}} A : B'$ implies $B \overset{*}{\longleftrightarrow}_{\beta} B'$.*

In the following, we will only consider functional PTS.

2.3. A scheme for λ-calculi with explicit substitutions

Because we do not want that our work depends on a particular calculus with explicit substitution, we use a scheme, as is done in [22]. Our scheme is slightly different, because we use named variables and precooking [13], so that the λ-term $\lambda x\, (y\, x)$ is translated into $\lambda(y\,[\text{shift}]\ \underline{1})$ instead of for instance $\lambda(\underline{2}\,\underline{1})$ using only De Bruijn indices. A pure term is therefore a term were free variables are correctly precooked. However, the results are the same in the two settings, provided we add conditions 10 and 11 in Definition 4.

Definition 3 (Substitution calculus). *A substitution calculus consists of a signature upon two sorts $\mathfrak{T}$ and $\mathfrak{S}$ which contains a constant $\text{shift} : \mathfrak{S}$, three function symbols $\lambda : \mathfrak{T} \hookrightarrow \mathfrak{T}$, $\cdot\, \cdot : \mathfrak{T} \times \mathfrak{T} \hookrightarrow \mathfrak{T}$ and $\cdot[\cdot] : \mathfrak{T} \times \mathfrak{S} \hookrightarrow \mathfrak{T}$, and in which can be defined two functions $\text{lift} : \mathfrak{S} \hookrightarrow \mathfrak{S}$ and $\text{cons} : \mathfrak{T} \hookrightarrow \mathfrak{S}$; and a sort-preserving rewrite system containing*

$$(ab)\,[s] \to a\,[s]\ b\,[s] \tag{App}$$

and

$$(\lambda a)\,[s] \to \lambda\,(a\,[\text{lift}(s)]) \tag{Lambda}$$

We also need the rule which triggers the β-reduction:

$$(\lambda a)b \to a\,[\text{cons}(b)] \tag{Beta}$$

It is *not* part of the rewrite system W of the substitution calculus. We denote by λ_W the system $W + (Beta)$.

Definition 4 (Scheme). *A substitution calculus W obeys the scheme if*

3

Empty $$\frac{}{[]\ \text{well-formed}}$$

Declaration $$\frac{\Gamma \vdash_{\text{PTS}} A : s}{\Gamma[x:A]\ \text{well-formed}}\quad s \in \mathcal{S}\ \text{and}\ x\ \text{not in}\ \Gamma$$

Sort $$\frac{\Gamma\ \text{well-formed}}{\Gamma \vdash_{\text{PTS}} s_1 : s_2}\quad \langle s_1, s_2 \rangle \in \mathcal{A}$$

Variable $$\frac{\Gamma\ \text{well-formed} \qquad x : A \in \Gamma}{\Gamma \vdash_{\text{PTS}} x : A}$$

Product $$\frac{\Gamma \vdash_{\text{PTS}} A : s_1 \qquad \Gamma[x:A] \vdash_{\text{PTS}} B : s_2}{\Gamma \vdash_{\text{PTS}} \Pi x : A\ B : s_3}\quad \langle s_1, s_2, s_3 \rangle \in \mathcal{R}$$

Application $$\frac{\Gamma \vdash_{\text{PTS}} T : \Pi x : A\ B \qquad \Gamma \vdash_{\text{PTS}} U : A}{\Gamma \vdash_{\text{PTS}} (T\ U) : \{U/x\}B}$$

Abstraction $$\frac{\Gamma \vdash_{\text{PTS}} \Pi x : A\ B : s \qquad \Gamma[x:A] \vdash_{\text{PTS}} T : B}{\Gamma \vdash_{\text{PTS}} \lambda x : A\ T : \Pi x : A\ B}$$

Conversion $$\frac{\Gamma \vdash_{\text{PTS}} T : A \qquad \Gamma \vdash_{\text{PTS}} B : s}{\Gamma \vdash_{\text{PTS}} T : B}\quad s \in \mathcal{S}\ \text{and}\ A \xleftrightarrow{\ *\ }_\beta B$$

Figure 2. Typing rules of a PTS $\langle \mathcal{S}, \mathcal{A}, \mathcal{R} \rangle$

1. *W is confluent*
2. *W is strongly normalizing*
3. *W-normal forms of terms are pure terms*
4. *$W(ab) = W(a)W(b)$ and $W(\lambda a) = \lambda W(a)$, where $W(a)$ denotes the W-normal form of a*
5. *for every substitution $s : \mathfrak{S}$ in W, $\underline{1}[\text{lift}(s)] \xleftrightarrow{\ *\ }_W \underline{1}$*
6. *for every substitution $s : \mathfrak{S}$ in W and $m \geq 1$, $\underline{m+1}[\text{lift}(s)] \xleftrightarrow{\ *\ }_W \underline{m}[s][\text{shift}]$*
7. *for every term $k : \mathfrak{T}$ and $m \geq 1$, $\underline{m+1}[\text{cons}(k)] \xleftrightarrow{\ *\ }_W \underline{m}$*
8. *for every term $k : \mathfrak{T}$, $\underline{1}[\text{cons}(k)] \xleftrightarrow{\ *\ }_W k$*
9. *for every m, $\underline{m}[\text{shift}] \xleftrightarrow{\ *\ }_W \underline{m+1}$*
10. *for every named variable x, substitution $s : \mathfrak{S}$, and $n \geq 0$, $x[\text{shift}]^{n+1}[\text{lift}(s)] \xleftrightarrow{\ *\ }_W x[\text{shift}]^n[s][\text{shift}]$*
11. *for every named variable x, term $k : \mathfrak{T}$, and $n \geq 0$, $x[\text{shift}]^{n+1}[\text{cons}(k)] \xleftrightarrow{\ *\ }_W x[\text{shift}]^n$*
12. *for every function symbol $\xi : \mathfrak{K}_1 \times \cdots \times \mathfrak{K}_q \to \mathfrak{S}$ and $m \geq 1$, one of the two following conditions holds:*
    - *there exists some n, distinct indices $i_1, \ldots, i_p$ in $\{1; \cdots; q\}$ and substitutions $u_1, \ldots, u_k$ such that for all $s_1, \ldots, s_q$ we have $\underline{m}[\xi(q_1, \ldots, q_n s)] \xleftrightarrow{\ *\ }_W \underline{n}[s_{i_1}] \ldots [s_{i_p}][u_1] \ldots [u_k]$;*
    - *there exists $i \in \{1; \cdots; q\}$ such that for all $s_1, \ldots, s_q$ we have $\underline{m}[\xi(q_1, \ldots, q_n s)] \xleftrightarrow{\ *\ }_W s_i$.*

In particular, these conditions implies that λ_W is confluent [22, Theorem 5.18]. Many calculi with explicit substitution are instances of this scheme, in particular λ_σ [1] (see [22] for others).

Note that we do not use the same parenthesis conventions for PTS terms and (first-order) terms of the scheme, in order to make a visual distinction between them.

3. Superdeductive systems for intuitionistic logic

3.1. Supernatural deduction

Supernatural deduction LJ^+ was introduced by Wack [27] as a complement to deduction modulo [15]. It con-

Introduction rules:

$$\frac{\Gamma \vdash A}{\Gamma \vdash A^*}$$

$$\frac{}{\Gamma \vdash \top^*}$$

$$\frac{\Gamma, P \vdash Q^*}{\Gamma \vdash (P \Rightarrow Q)^*}$$

$$\frac{\Gamma \vdash P^* \qquad \Gamma \vdash Q^*}{\Gamma \vdash (P \wedge Q)^*}$$

$$\frac{\Gamma \vdash P^*}{\Gamma \vdash (\forall x.\ P)^*}\ x\ \text{not free in}\ \Gamma$$

Elimination rules:

$$\frac{\Gamma \vdash A^*}{\Gamma \vdash A}$$

$$\frac{\Gamma \vdash \top^*}{\Gamma \vdash \top}$$

$$\frac{\Gamma \vdash (P \Rightarrow Q)^* \qquad \Gamma \vdash P}{\Gamma \vdash Q^*}$$

$$\frac{\Gamma \vdash (P \wedge Q)^*}{\Gamma \vdash P^*}\qquad \frac{\Gamma \vdash (P \wedge Q)^*}{\Gamma \vdash Q^*}$$

$$\frac{\Gamma \vdash (\forall x.\ P)^*}{\Gamma \vdash (\{t/x\}P)^*}$$

Figure 3. Rules for computing super introduction and elimination rules

sists in computing new inference rules that are adapted to the theory we are considering. For technical reasons, this theory has to be expressed as two rewrite systems. The former rewrites terms into terms, the latter rewrites atomic propositions to arbitrary first-order formulæ. The idea is to shorten the proofs by precomputing the steps that naturally occur after the unfolding of a definition corresponding to the proposition rewrite system.

Supernatural deduction is based on natural deduction [18] and the new rules can be seen as introduction and elimination rules corresponding to the rewrite rule. Given a rewrite rule $A \to P$, its introduction rule is computed by applying from bottom to top all the introduction rules of figure 3 on P^* while a $*$ remains. The premises of the new rule are then the remaining open leaves, whereas its conclusion is A, and we keep all the freshness conditions. Elimination rules are computed using the elimination rules of figure 3 from top left to bottom. All rules are then applied modulo the term rewrite system (see [15]).

Example 1*:* Let $\mathcal{C}rab$ be the rule $A \to B \wedge (A \Rightarrow C)$.

The computation of the introduction rule is

$$\dfrac{\Gamma \vdash B^* \qquad \dfrac{\Gamma, A \vdash C^*}{\Gamma \vdash (A \Rightarrow C)^*}}{\Gamma \vdash (B \wedge (A \Rightarrow C))^*}$$

and the new rule is therefore

$$\mathcal{C}rab_I \ \dfrac{\Gamma \overset{+}{\vdash} B \qquad \Gamma, A \overset{+}{\vdash} C}{\Gamma \overset{+}{\vdash} A}$$

The computation of the elimination rules are

$$\dfrac{\Gamma \vdash (B \wedge (A \Rightarrow C))^*}{\Gamma \vdash B^*} \qquad \dfrac{\dfrac{\Gamma \vdash (B \wedge (A \Rightarrow C))^*}{\Gamma \vdash (A \Rightarrow C)^*} \qquad \Gamma \vdash A}{\Gamma \vdash C^*}$$

and the new rules are therefore

$$\mathcal{C}rab_{E1} \ \dfrac{\Gamma \overset{+}{\vdash} A}{\Gamma \overset{+}{\vdash} B} \quad \text{and} \quad \mathcal{C}rab_{E2} \ \dfrac{\Gamma \overset{+}{\vdash} A \qquad \Gamma \overset{+}{\vdash} A}{\Gamma \overset{+}{\vdash} C}$$

3.2. Super sequent calculus

The super sequent calculus was introduced by Brauner, Houtmann and Kirchner [8] as the analog of supernatural deduction for the sequent calculus for classical logic. We propose here its counterpart for intuitionistic logic. It is based on LB (see Section 2.1). The idea to compute the left (resp. right) rule for a rewrite rule $A \to P$ is to apply LB rules on $\Gamma, P^* \vdash \Delta$ (resp. $\Gamma \vdash P^*, \Delta$) unless there is a permutation problem. Rules for the computation of super sequent rules LB^+ are summed up in figure 4. Once again, all inference rules are applied modulo the term rewrite system.

Example 2*:* We consider the rewrite rule $\mathcal{C}rab$ of Example 1. The computation of the left rule is

$$\dfrac{\dfrac{\Gamma, B, C^* \vdash \Delta \qquad \Gamma, B \vdash A^*, \Delta}{\Gamma, B^*, (A \Rightarrow C)^* \vdash \Delta}}{\Gamma, (B \wedge (A \Rightarrow C))^* \vdash \Delta}$$

and the new rule is therefore

$$\mathcal{C}rab_L \ \dfrac{\Gamma, B, C \overset{+}{\vdash} \Delta \qquad \Gamma, B \overset{+}{\vdash} A, \Delta}{\Gamma, A \overset{+}{\vdash} \Delta}$$

The computation of the right rule is

$$\dfrac{\Gamma \vdash B^*, \Delta \qquad \dfrac{\Gamma, A \vdash C^*}{\Gamma \vdash (A \Rightarrow C)^*, \Delta}}{\Gamma \vdash (B \wedge (A \Rightarrow C))^*, \Delta}$$

and the new rule is therefore

$$\mathcal{C}rab_R \ \dfrac{\Gamma \overset{+}{\vdash} B, \Delta \qquad \Gamma, A \overset{+}{\vdash} C}{\Gamma \overset{+}{\vdash} A, \Delta}$$

Left rules:

$$\dfrac{\Gamma, A \vdash \Delta}{\Gamma, A^* \vdash \Delta}$$

$$\dfrac{\Gamma, \top \vdash \Delta}{\Gamma, \top^* \vdash \Delta}$$

$$\dfrac{\Gamma, Q^* \vdash \Delta \qquad \Gamma \vdash P, \Delta}{\Gamma, (P \Rightarrow Q)^* \vdash \Delta}$$

$$\dfrac{\Gamma, P^*, Q^* \vdash \Delta}{\Gamma, (P \wedge Q)^* \vdash \Delta}$$

$$\dfrac{\Gamma, (\{t/x\}P)^* \vdash \Delta}{\Gamma, (\forall x.\ P)^* \vdash \Delta}$$

Right rules

$$\dfrac{\Gamma \vdash A, \Delta}{\Gamma \vdash A^*, \Delta}$$

$$\dfrac{}{\Gamma \vdash \top^*, \Delta}$$

$$\dfrac{\Gamma, P \vdash Q^*}{\Gamma \vdash (P \Rightarrow Q)^*, \Delta}$$

$$\dfrac{\Gamma \vdash P^*, \Delta \qquad \Gamma \vdash Q^*, \Delta}{\Gamma \vdash (P \wedge Q)^*, \Delta}$$

$$\dfrac{\Gamma \vdash P^*}{\Gamma \vdash (\forall x.\ P)^*, \Delta} \ x \text{ not free in } \Gamma$$

Figure 4. Rules for computing super left and right rules

We also introduce the notion of permutation-problem free formulæ, for which we will prove that the atomic super sequent calculus is complete:

Definition 5 (Permutation-problem freeness)**.** *The set of permutation-problem free formulæ (ppf formulæ) is build on the the following grammar:*

$$P \overset{\text{def}}{=} A \mid \top \mid A \Rightarrow P \mid P \wedge P \mid \forall x.\ P$$

where A denotes an atomic proposition.

A rewrite rule is said to be ppf *if its right-hand side is ppf, and so does a rewrite system if all its rules are.*

Proposition 3 (Atomic LB^+)**.** *If the rewrite system is ppf, and Γ and Δ contains only atomic formulæ, then $\Gamma \vdash \Delta$ is provable iff it is provable with only identity and super rules.*

Proof. The idea is to use a procedure similar to the cut elimination, which transforms cuts around a formula with a connector, into cuts around subformulæ. This shows that we only need atomic cuts. Then, remark that the rewrite system is ppf iff the new left and right rules contains only atomic propositions. Thus, starting with atomic propositions, we can only apply atomic cuts or super rules, and these do not introduce non-atomic propositions. More details can be found in Appendix A.1. $\square$

3.3. Translations between these systems

The proofs of this section can be found in Appendix A.

THEOREM 4.

For a given rewrite system, any sequent provable in NJ^+ can be proved in LB^+.

Corollary 5. *A sequent is provable in* LB^+ *if it is provable in* LB *adding a compatible theory in the sense of [15, Definition 1.4] in the hypotheses.*

Proof. Wack [27] proved that NJ^+ is equivalent to NJ with assumptions in the compatible theory, and we can translate every proof of LB into NJ, as well as proofs in NJ^+ into proofs of LB^+. $\qquad\square$

Note: LB^+ without the cut rule may be incomplete in that sense, as shown by the example of the *Crab* rule. In LB^+ there is no cut-free proof of $B \vdash^+ C$ but we can build the first proof of figure 5 with cuts.

THEOREM 6.
A proof of a sequent in LB^+ *using only identity and super rules can be transformed into a proof of the same sequent in* NJ^+ *using only super rules and* Axiom.

Corollary 7. *For ppf rewrite systems, sequents containing only atomic propositions can be proved in* NJ^+ *iff they do using only super rules and* Axiom.

Note: The admissibility of the cut rule in the super sequent calculus is not needed for the completeness of the atomic supernatural deduction. For instance, for the rewrite rule *Crab*, the first proof of $B \vdash^+ C$ in LB^+ in figure 5 is translated into the second proof in NJ^+ which indeed contains only atomic formulæ.

4. PTS as superdeductive theories

In this section, we consider some functional PTS $\langle \mathcal{S}, \mathcal{A}, \mathcal{R}\rangle$, we define its corresponding superdeductive system and prove the soundness and conservativeness of the translation.

4.1. The system

The term rewrite rules correspond to some λ-calculus with explicit substitutions λ_W. As we do not want to rely on a particular calculus, we use the scheme proposed by Kesner [22]. See Section 2.3.

We also use some function symbols to represent the Π binder of the PTS. As done in [11], we need a new function symbol $\dot{\pi}_{\langle s_1,s_2,s_3\rangle}$ for each rule $\langle s_1,s_2,s_3\rangle \in \mathcal{R}$. We need to tell how they interact with explicit substitutions by adding term rewrite rules:

$$\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b)[s] \to \dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a[s],b[\mathsf{lift}(s)]) \ .$$

Note that we use lift on the second argument as is done for λ in (Lambda), because some variable is bound by $\dot{\pi}_{\langle s_1,s_2,s_3\rangle}$.

We also need a constant s for each sort $s \in \mathcal{S}$. We add the following term rewrite rules, which permit to avoid precooking the s constants in the translation from PTS terms to

$$(2)_{E1} \ \frac{\Gamma \vdash^+ \varepsilon\left(\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b),\mathsf{s}_3\right)}{\Gamma \vdash^+ \varepsilon(a,\mathsf{s}_1)}$$

$$(2)_{E2} \ \frac{\Gamma \vdash^+ \varepsilon\left(\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b),\mathsf{s}_3\right) \qquad \Gamma \vdash^+ \varepsilon(u,a)}{\Gamma \vdash^+ \varepsilon(b[\mathsf{cons}(u)],\mathsf{s}_2)}$$

$$(3)_{E1} \ \frac{\Gamma \vdash^+ \varepsilon\left(t,\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b)\right)}{\Gamma \vdash^+ \varepsilon\left(\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b),\mathsf{s}_3\right)}$$

$$(3)_{E2} \ \frac{\Gamma \vdash^+ \varepsilon\left(t,\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b)\right) \qquad \Gamma \vdash^+ \varepsilon(u,a)}{\Gamma \vdash^+ \varepsilon(t\,u,b[\mathsf{cons}(u)])}$$

Figure 7. Elimination rules of the NJ^+ **systems for the PTS**

first-order terms (see below):

$$\mathsf{s}[t] \to \mathsf{s} \ .$$

In the following, $\longrightarrow$ designates the rewrite relation associated with the term rewrite system we have just defined.

Let $\varepsilon : \mathfrak{T} \times \mathfrak{T}$ be a predicate. $\varepsilon(a,b)$ should be seen as a has type b. The proposition rewrite rules permits to simulate the inference rules of the PTS:

$$\varepsilon(\mathsf{s}_1,\mathsf{s}_2) \ \to \ \top \qquad\qquad (\langle s_1,s_2\rangle \in \mathcal{A}) \ (1)$$
$$\varepsilon\left(\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b),\mathsf{s}_3\right) \ \to \ \varepsilon(a,\mathsf{s}_1) \wedge \qquad\qquad\qquad (2)$$
$$\forall z.\ \varepsilon(z,a) \Rightarrow \varepsilon(b[\mathsf{cons}(z)],\mathsf{s}_2)$$
$$\varepsilon\left(t,\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b)\right) \ \to \ \varepsilon\left(\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b),\mathsf{s}_3\right) \wedge \qquad (3)$$
$$\forall z.\ \varepsilon(z,a) \Rightarrow \varepsilon(t\,z,b[\mathsf{cons}(z)])$$

These rewrite rules leads to the inference rules in supernatural deduction represented in figures 6 and 7.

We can now define a translation from PTS terms (i.e. well-typed pseudo terms) into our first-order terms. This translation is inspired from the one between λ-calculus with de Bruijn indices and λ_σ with precooking [13, Definition 4.1], except that we translate directly from a λ-calculus with names, and we deal with the binder Π. $\overline{\ell}$ means the length of the list ℓ in the usual sense.

Definition 6 (Translation). *The translation takes as parameters a context* Γ *and a list* ℓ *(which are omitted if unchanged):*

$$
\begin{aligned}
|x|_{\ell_1::x::\ell_2} &\stackrel{\text{def}}{=} \overline{\ell_1 + 1} & \text{if } x \notin \ell_1 \\
|x|_\ell &\stackrel{\text{def}}{=} x[\mathsf{shift}]^{\overline{\ell}} & \text{if } x \notin \ell \\
|s|_\ell &\stackrel{\text{def}}{=} \mathsf{s} & (s \in \mathcal{S}) \\
|\lambda x : A\ T|_\ell^\Gamma &\stackrel{\text{def}}{=} \lambda\, |T|_{x::\ell}^{\Gamma[x:A]} \\
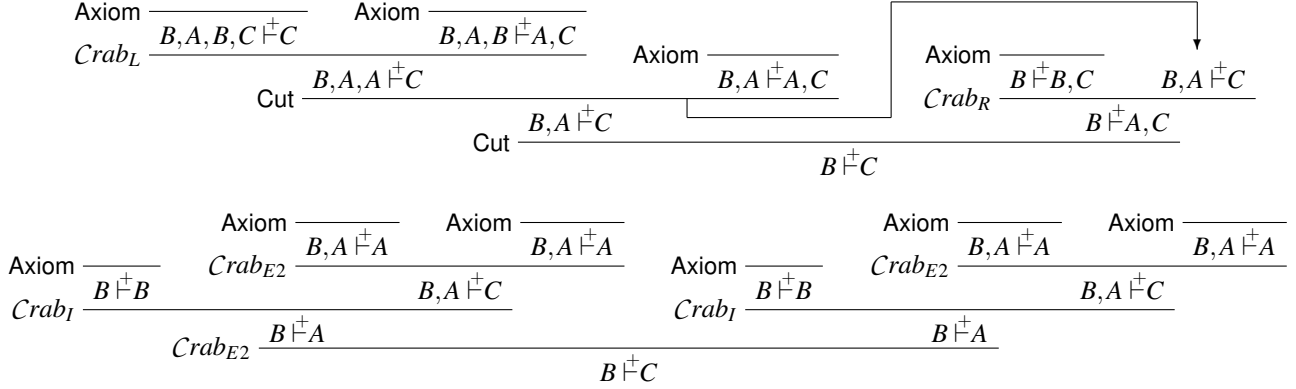|A\ B| &\stackrel{\text{def}}{=} |A|\ |B|
\end{aligned}
$$

$$\mathcal{C}rab_L \dfrac{\text{Axiom } \dfrac{}{B,A,B,C \vdash^{+} C} \quad \text{Axiom } \dfrac{}{B,A,B \vdash^{+} A,C}}{\text{Cut } \dfrac{B,A,A \vdash^{+} C}{\text{Cut } \dfrac{B,A \vdash^{+} C}{B \vdash^{+} C}}} \quad \text{Axiom } \dfrac{}{B,A \vdash^{+} A,C} \quad \mathcal{C}rab_R \dfrac{\text{Axiom } \dfrac{}{B \vdash^{+} B,C} \quad B,A \vdash^{+} C}{B \vdash^{+} A,C}$$

$$\mathcal{C}rab_I \dfrac{\text{Axiom } \dfrac{}{B \vdash^{+} B} \quad \mathcal{C}rab_{E2} \dfrac{\text{Axiom } \dfrac{}{B,A \vdash^{+} A} \quad \text{Axiom } \dfrac{}{B,A \vdash^{+} A}}{B,A \vdash^{+} C}}{\mathcal{C}rab_{E2} \dfrac{B \vdash^{+} A}{B \vdash^{+} C}} \qquad \mathcal{C}rab_I \dfrac{\text{Axiom } \dfrac{}{B \vdash^{+} B} \quad \mathcal{C}rab_{E2} \dfrac{\text{Axiom } \dfrac{}{B,A \vdash^{+} A} \quad \text{Axiom } \dfrac{}{B,A \vdash^{+} A}}{B,A \vdash^{+} C}}{B \vdash^{+} A}$$

Figure 5. Example of the non-admissibility of cut in LB^{+} and its translation in NJ^{+}

$$(1)_I \ \dfrac{}{\Gamma \vdash^{+} \varepsilon(\mathsf{s}_1,\mathsf{s}_2)} \ \langle s_1,s_2 \rangle \in \mathcal{A} \qquad\qquad (2)_I \ \dfrac{\Gamma \vdash^{+} \varepsilon(a,\mathsf{s}_1) \quad \Gamma, \varepsilon(z,a) \vdash^{+} \varepsilon(b\,[\mathsf{cons}(z)],\mathsf{s}_2)}{\Gamma \vdash^{+} \varepsilon\left(\dot{\pi}_{\langle s_1,s_2,s_3 \rangle}(a,b),\mathsf{s}_3\right)} \ z \text{ not free in } \Gamma, \ \langle s_1,s_2,s_3 \rangle \in \mathcal{R}$$

$$(3)_I \ \dfrac{\Gamma \vdash^{+} \varepsilon\left(\dot{\pi}_{\langle s_1,s_2,s_3 \rangle}(a,b),\mathsf{s}_3\right) \quad \Gamma, \varepsilon(z,a) \vdash^{+} \varepsilon(t\,z,b\,[\mathsf{cons}(z)])}{\Gamma \vdash^{+} \varepsilon\left(t,\dot{\pi}_{\langle s_1,s_2,s_3 \rangle}(a,b)\right)} \ z \text{ not free in } \Gamma, \ \langle s_1,s_2,s_3 \rangle \in \mathcal{R}$$

Figure 6. Introduction rules of the NJ^{+} systems for the PTS

$$|\Pi x : A\ B|_\ell^\Gamma \ \overset{\text{def}}{\equiv} \ \dot{\pi}_{\langle s_1,s_2,s_3 \rangle}\left(|A|_\ell^\Gamma, |B|_{x::\ell}^{\Gamma[x:A]}\right)$$

where s_1 corresponds the type of A in Γ, s_2 the type of B in $\Gamma[x:A]$, and $\langle s_1,s_2,s_3 \rangle \in \mathcal{R}$.

Then, a context is translated inductively by $|[]| \overset{\text{def}}{=} \emptyset$ and $|\Gamma[x:A]| \overset{\text{def}}{=} |\Gamma|, \varepsilon\left(x,|A|_{[]}^\Gamma\right)$, and a typing judgment $\Gamma \vdash_{\text{PTS}} A : B$ is translated into $|\Gamma| \vdash^{+} \varepsilon\left(|A|_{[]}^\Gamma, |B|_{[]}^\Gamma\right)$. In the following, contexts and lists will be omitted if they can be deduce from the environment.

Note that this translation is deterministic, because the PTS is assumed to be functional. Note also that if $\Gamma \subseteq \Gamma'$ then $|T|^\Gamma = |T|^{\Gamma'}$ if they are defined, by [4, Lemma 5.2.12].

4.2. Soundness and conservativeness

We need the following properties to prove the soundness and the conservativeness of the NJ^{+} system w.r.t. the PTS. Their proofs can be found in Appendix B.

Lemma 8.
- $|B|_x \left[\mathsf{cons}(|U|_{[]})\right] \longleftrightarrow^{*} |\{U/x\}B|_{[]}$.

- *If $|A| \overset{*}{\longrightarrow} a$ then there exists A' such that $a \longrightarrow^{*} |A'|$.*

- $|B|_z = \{\underline{1}/z\,[\mathsf{shift}]\}(|B|_{[]}\,[\mathsf{shift}])$

- *If $\langle s_1,s_2,s_3 \rangle \in \mathcal{R}$, $\Gamma \vdash_{\text{PTS}} A : s_1$ and $\Gamma[x:A] \vdash_{\text{PTS}} B : s_2$, x is not free in b, $a \overset{*}{\longrightarrow} |A|^\Gamma$ and $b\,[\mathsf{cons}(x)] \longrightarrow |B|^{\Gamma[x:A]}$ then $\dot{\pi}_{\langle s_1,s_2,s_3 \rangle}(a,b) \longleftrightarrow^{*} |\Pi x : A\ B|^\Gamma$.*

- *If $\dot{\pi}_{\langle s_1,s_2,s_3 \rangle}(a,b) \overset{*}{\longrightarrow} |C|$ then there exists A, B such that $C = \Pi x : A\ B$.*

THEOREM 9 (Soundness).
For all PTS terms T and A, if $\Gamma \vdash_{\text{PTS}} T : A$ then $|\Gamma| \vdash^{+} \varepsilon(|T|,|A|)$.

Proof. The typing rules of the PTS, represented on Figure 2, are translated into the corresponding derivations in NJ^{+} represented on Figure 8. We add some comments:

The $\equiv$ derivations are not really inference rules, but corresponds to the fact that we are working modulo the term rewrite system λ_W; they permit to let the computation appear. Γ' in the translation of Abstraction is $|\Gamma|, \varepsilon(z,|A|)$. In the translation of Conversion, we do not check the type of the converts; this is not problematic as shown by the conservativeness result below. In the translation of Variable, $\varepsilon(x,|A|)$ is in $|\Gamma|$ by definition of the translation of a context. In the translations of Abstraction and Product, we use the fact that z is free in $|\Gamma|$ to translate proofs of $\Gamma[x:A] \vdash_{\text{PTS}} \alpha$ into proofs of $\Gamma[z:A] \vdash_{\text{PTS}} \{z/x\}\alpha$. In the translation of Abstraction, we know that $|s| = \mathsf{s}_3$ by Lemma 2. $\square$

$$(2)_I \quad \cfrac{|\Gamma| \vdash^{+}\varepsilon(|A|,\mathsf{s_1}) \qquad \cfrac{\equiv \cfrac{|\Gamma|,\varepsilon(z,|A|) \vdash^{+}\varepsilon(|\{z/x\}B|,\mathsf{s_2})}{|\Gamma|,\varepsilon(z,|A|) \vdash^{+}\varepsilon(|B|_x[\mathsf{cons}(z)],\mathsf{s_2})}\ \text{Lem. 8}}{}\ z \text{ not free in } |\Gamma|}{|\Gamma| \vdash^{+}\varepsilon\big(\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(|A|,|B|_x),\mathsf{s_3}\big)}$$

Empty : ⎫
    ⎬ we do not need to look at the wellformedness of the context
Declaration : ⎭

$$(3)_{E2} \quad \cfrac{|\Gamma| \vdash^{+}\varepsilon\big(|T|,\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(|A|,|B|_x)\big) \qquad |\Gamma| \vdash^{+}\varepsilon(|U|,|A|)}{\equiv \cfrac{|\Gamma| \vdash^{+}\varepsilon(|T|\ |U|,|B|_x[\mathsf{cons}(|U|)])}{|\Gamma| \vdash^{+}\varepsilon(|T|\ |U|,|\{U/x\}B|)}\ \text{Lem. 8}}$$

$$(1)_I \quad \cfrac{}{\Gamma \vdash^{+}\varepsilon(\mathsf{s_1},\mathsf{s_2})}\ \langle s_1,s_2\rangle \in \mathcal{A}$$

$$(3)_I \quad \cfrac{|\Gamma| \vdash^{+}\varepsilon\big(\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(|A|,|B|_x),\mathsf{s_3}\big) \qquad \cfrac{\equiv \cfrac{\Gamma' \vdash^{+}\varepsilon(|\{z/x\}T|,|\{z/x\}B|)}{\Gamma' \vdash^{+}\varepsilon((\lambda|T|_x)z,|B|_x[\mathsf{cons}(z)])}\ \begin{array}{l}(\lambda|T|_x)z \xrightarrow{*} \\ |T|_x[\mathsf{cons}(z)]\end{array}}{}\ z \text{ not free in } |\Gamma|}{|\Gamma| \vdash^{+}\varepsilon\big(\lambda|T|_x,\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(|A|,|B|_x)\big)}$$

Axiom $\quad \cfrac{}{|\Gamma| \vdash^{+}\varepsilon(x,|A|)}$

Conversion: using λ_W in the modulo

Figure 8. Derivations in NJ^{+} corresponding to the typing rules of a PTS (see Figure 2)

THEOREM 10 (Conservativeness).
For all well-formed context Γ, for all terms a, b if $|\Gamma| \vdash^{+}\varepsilon(a,b)$ then there exists A and B such that $a \xrightarrow{*} |A|^\Gamma$, $b \xrightarrow{*} |B|^\Gamma$ and $\Gamma \vdash_{\text{PTS}} A : B$.

Proof. By induction on the proof of $|\Gamma| \vdash^{+}\varepsilon(a,b)$ and the number of uses of the modulo on terms.

If the last rule is applied modulo the term rewrite system, we apply it without modulo and then consider the modulo:

$$\equiv \cfrac{|\Gamma| \vdash^{+}\varepsilon(a',b') \qquad a \xleftrightarrow{*} a'}{|\Gamma| \vdash^{+}\varepsilon(a,b) \qquad b \xleftrightarrow{*} b'}$$

By induction hypothesis, there exists some A' and B' such that $a' \xrightarrow{*} |A'|$, $b' \xrightarrow{*} |B'|$ and $\Gamma \vdash_{\text{PTS}} A' : B'$. By confluence of λ_W, there exists a'' such that $a \xrightarrow{*} a'' \xleftarrow{*} |A'|$. By Lemma 8 there exists some A such that $a'' \xrightarrow{*}_W |A|$. By subject reduction, $\Gamma \vdash_{\text{PTS}} A : B'$. By confluence of λ_W, there exists b'' such that $b \xrightarrow{*} b'' \xleftarrow{*} |B'|$. By Lemma 8 there exists B such that $b'' \xrightarrow{*} |B|$. By Lemma 1, either B' is a sort, in which case it is in normal form and $B = B'$, or there exists some sort $s \in \mathcal{S}$ such that $\Gamma \vdash_{\text{PTS}} B' : s$, and by subject reduction $\Gamma \vdash_{\text{PTS}} B : s$. Therefore, using Conversion we deduce that $\Gamma \vdash_{\text{PTS}} A : B$. In the following we can therefore assume that the inference rules are applied without modulo. We use the names of the variables used on Fig. 6 and 7.

- If the last rule is an axiom, hence of the form:

Axiom $\quad \cfrac{}{|\Gamma_1|,\varepsilon(x,b),|\Gamma_2| \vdash^{+}\varepsilon(x,b)}$

By definition of the translation of contexts, this means that $b = |B|$ with $x : B \in \Gamma$. Because by assumption Γ is well-formed, we can therefore use Variable to get a proof of $\Gamma \vdash_{\text{PTS}} x : B$.

- If the last rule is $(1)_I$: as Γ is assumed to be well-formed, we can apply Sort.

- If the last rule is $(2)_I$: By induction hypothesis on the left subproof, there exists A such that $a \xrightarrow{*} |A|$ and $\Gamma \vdash_{\text{PTS}} A : s$. As z is free in Γ, $\Gamma[z : A]$ is well-formed. Therefore we can use the induction hypothesis on the right subproof: there exists B such that $b[\mathsf{cons}(z)] \xrightarrow{*} |B|$ and $\Gamma[z : A] \vdash_{\text{PTS}} B : s_2$. We can consider that z is free in b, hence by Lemma 8 $\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b) \xleftrightarrow{*} |\Pi z : A\ B|$, so we can conclude using the confluence of λ_W, subject reduction and Product.

- If the last rule is $(3)_I$: By induction hypothesis on the left subproof, there exists C such that $\Gamma \vdash_{\text{PTS}} C : s_3$ and $\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b) \xrightarrow{*} |C|$. By Lemma 8 there exists A, B_1 such that $C = \Pi x : A\ B_1$. Because λ_W cannot change the function symbols $\dot{\pi}_{\langle s_1,s_2,s_3\rangle}$, that means that $|C| = \dot{\pi}_{\langle s_1,s_2,s_3\rangle}(|A|,|B_1|_x)$, $a \xrightarrow{*} |A|$ and $b \xrightarrow{*} |B_1|_x$. Hence $\Gamma \vdash_{\text{PTS}} A : s_1$, and the context $\Gamma[z : A]$ is well-formed because z is free in $|\Gamma|$. By induction hypothesis on the right subproof, there exists T', B_2 such that $\Gamma[z : A] \vdash_{\text{PTS}} T' : B_2$ with $tz \xrightarrow{*} |T'|$ and $b[\mathsf{cons}(z)] \xrightarrow{*} |B_2|$. By Lemma 8 $|\{z/x\}B_1| \xleftrightarrow{*} |B_2|$, hence by confluence of λ_W and Lemma 8 there is a B such that $|B|$ is a common reduct. By α-conversion and subject reduction $\Gamma \vdash_{\text{PTS}} \Pi z : A\ B : s_3$ and using Conversion $\Gamma[z : A] \vdash_{\text{PTS}} T' : B$. Using Abstraction we obtain $\Gamma \vdash_{\text{PTS}} \lambda z : A\ T' : \Pi z : A\ B$. By Lemma 8, $\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b) \xrightarrow{*} \Pi z : A\ B$.

Now consider the reduction $tz \xrightarrow{*} |T'|$. There are two cases: Either the outermost application never reduces, and therefore $|T'| = t_1 z$. In that case, by definition of the translation, we must have $T' = (T\ z)$ for some T with $t \xrightarrow{*} |T|$. By subject reduction for η, from $\Gamma \vdash_{\text{PTS}} \lambda z : A\ (T\ z) : \Pi z : A\ B$ we

8

obtain $\Gamma \vdash_{\mathrm{PTS}} T : \Pi z : A\ B$.

In the other case, we have $tz \xrightarrow{*} (\lambda t_1)z \xrightarrow{*} |T'|$. Because z is not free in Γ, we can consider it is not free in t. Therefore

$$
\begin{aligned}
\lambda(t\,[\text{shift}]\ \underline{1}) &= \{\underline{1}/z\,[\text{shift}]\}\lambda(t\,[\text{shift}]\ z\,[\text{shift}])\\
&= \lambda(\{\underline{1}/z\,[\text{shift}]\}((tz)\,[\text{shift}]))\\
&\xrightarrow{*} \lambda\left(\{\underline{1}/z\,[\text{shift}]\}(|T'|_{[]}\,[\text{shift}])\right)\\
&\xleftrightarrow[\text{Lem. 8}]{*} \lambda|T'|_z = |\lambda z : A\ T'|
\end{aligned}
$$

But

$$
\begin{aligned}
\lambda(t\,[\text{shift}]\ \underline{1}) &\xrightarrow{*} \lambda((\lambda t_1)\,[\text{shift}]\ \underline{1})\\
&\xrightarrow{*} \lambda((\lambda t_1\,[\text{lift}(\text{shift})])\underline{1})\\
&\xrightarrow{(Beta)} \lambda(t_1\,[\text{lift}(\text{shift})]\,[\text{cons}(\underline{1})])\\
&\xrightarrow[{[22,\ \text{Lem. }4.5]}]{*} \lambda t_1 \xleftarrow{*} t \quad .
\end{aligned}
$$

Therefore $t \xleftrightarrow{*} |\lambda z : A\ T'|$ and by confluence of λ_W and Lemma 8, there exists T such that $t \xrightarrow{*} |T|$ and $\lambda z : A\ T' \xrightarrow{*}_\beta T$. We conclude by subject reduction.

• If the last rule is $(2)_{E1}$: By induction hypothesis, there exists T such that $\dot\pi_{\langle s_1, s_2, s_3\rangle}(a,b) \xrightarrow{*} |T|$ and $\Gamma \vdash_{\mathrm{PTS}} T : s_3$. By Lemma 8, there exists A, B such that $T = \Pi x : A\ B$ and $a \xrightarrow{*} |A|$. Then by definition of the translation, $|T| = \dot\pi_{\langle s_1', s_2', s_3'\rangle}(a',b')$ where s_1' is the type of A in Γ, which exists because $\Gamma \vdash_{\mathrm{PTS}} \Pi x : A\ B : s_3$. But the λ_W reductions cannot change the function symbol $\dot\pi_{\langle s_1, s_2, s_3\rangle}$, hence $s_1 = s_1'$ and $\Gamma \vdash_{\mathrm{PTS}} A : s_1$.

• If the last rule is $(2)_{E2}$ or $(3)_{E1}$: this comes from the definition of the translation, as in the previous case. We also need Substitution in Lemma 1 for $(2)_{E2}$.

• If the last rule is $(3)_{E3}$: Using the same techniques as in the previous cases, by induction hypothesis, we get proofs of $\Gamma \vdash_{\mathrm{PTS}} T : \Pi x : A\ B$ and $\Gamma \vdash_{\mathrm{PTS}} U : A$ for some A, B, T, U such that $a \xrightarrow{*} |A|$, $b \xrightarrow{*} |B|$, $t \xrightarrow{*} |T|$ and $u \xrightarrow{*} |U|$ Using Application we get $\Gamma \vdash_{\mathrm{PTS}} (T\ U) : \{U/x\}B$. We can conclude because $tu \xrightarrow{*} |(T\ U)|$ and by Corollary 8 $b[\text{cons}(u)] \xrightarrow{*} |\{U/x\}B|$.

• If the last rule is another (i.e. logical) rule: we do not need this case because of Corollary 7. □

Note: Because terms are β-convertible on the left of $\varepsilon(\cdot, \cdot)$ too, there is no correspondence between the term on the left of $\varepsilon(\cdot, \cdot)$ and the proof in NJ^+. Nevertheless, the former is β-convertible to a term corresponding to the latter. This also implies that we cannot hope for a stronger result, since for instance it is not possible to find accurate types to prove $\vdash_{\lambda\Pi}((\lambda x : ?\ \lambda y : ?\ y\ \lambda x : ?\ (x\ x))\ *) : \square$ in the PTS $\lambda\Pi$ whereas we can prove $\vdash \varepsilon((\lambda\lambda\underline{1})(\lambda\underline{1}\underline{1})\star, \square)$ in the corresponding supernatural deduction.

But, for Theorem 10, if the PTS is strongly normalizing, then A and B are too, so a and b are weakly normalizing and we can reach some $|A'|$ and $|B'|$ from a and b such that $\Gamma \vdash_{\mathrm{PTS}} A' : B'$ by simulating leftmost β-reduction.

5. Applications

5.1. Using λ_σ for the explicit-substitution calculus

Because λ_σ [1] is an instance of the scheme (see [22]), we can use it as λ-calculus with explicit substitutions. If we do so, Lemma 8 becomes

Lemma 11. $|B|_x \left[|U|_{[]} \cdot id\right] \xrightarrow{*} |\{U/x\}B|_{[]}$

It means that in the proof of soundness, rewriting is performed from bottom to top after an introduction rule, and from top to bottom after an elimination rules. If we translate such a proof into LB^+ using Theorem 4, terms are rewritten only from the bottom to the top of the proof. This corresponds to the asymmetric sequent calculus modulo of Dowek [12]. This is also what is actually implemented in the proof assistant based on superdeduction Lemuridæ [8], where terms can only be reduced when building the proof tree from bottom to top.

5.2. A sequent calculus for Pure Types Systems

By applying Theorem 4 we obtain a sequent calculus presentation of Pure Type Systems. We do not detail all super left and right rules—that can be computed by the reader using Figure 4—but only the rules for (3):

$$
\Gamma, \varepsilon\left(\dot\pi_{\langle s_1, s_2, s_3\rangle}(a,b), \mathsf{s_3}\right), \varepsilon(t\ u, b[\text{cons}(u)]) \vdash^{+} \Delta
$$

$$
(3)_L \ \frac{\Gamma, \varepsilon\left(\dot\pi_{\langle s_1, s_2, s_3\rangle}(a,b), \mathsf{s_3}\right) \vdash^{+} \varepsilon(u,a), \Delta}{\Gamma, \varepsilon\left(t, \dot\pi_{\langle s_1, s_2, s_3\rangle}(a,b)\right) \vdash^{+} \Delta}
$$

$$
\Gamma \vdash^{+} \varepsilon\left(\dot\pi_{\langle s_1, s_2, s_3\rangle}(a,b), \mathsf{s_3}\right), \Delta
$$

$$
(3)_R \ \frac{\Gamma, \varepsilon(z,a) \vdash^{+} \varepsilon(t\ z, b[\text{cons}(z)])}{\Gamma \vdash^{+} \varepsilon\left(t, \dot\pi_{\langle s_1, s_2, s_3\rangle}(a,b)\right), \Delta}
$$

We can see how $(3)_L$ can be used to apply a function t to an argument u.

Sequent calculi for PTS were already introduced by Gutiérrez and Ruiz [20] and by Lengrand *et al.* [23]. It remains to be investigated how the superdeductive system relates to these, what does not seem trivial.

6. Conclusion

We have defined a natural encoding from functional PTS into supernatural deduction. We can therefore characterize PTS as first-order theories. We have proved the correctness and conservativeness of the translation, leading to a correspondence between valid typing judgments in the PTS and β-reducts of provable sequents in supernatural deduction.

We also have proved, through an equivalence with the super sequent calculus, that we only need the newly computed rules to show sequents consisting only of atomic propositions. This is a general result which shows how superdeduction can be used as a logical framework by encapsulating the propositions of the system we want to encode into ε constructs. Having encoded all reasonable PTS proves how powerful this can be. It is worth to note how close the superdeductive system is to the PTS. We also obtain sequent calculus presentations of PTS, which are to be compared to existing ones [20, 23].

We can then use this encoding to check proofs produced by proof assistants based on PTS. To do so, we could, because of the equivalence with first-order logic without modulo but within a compatible theory $\mathcal{T}$, use an intuitionistic first-order theorem prover. To prove a theorem A, we have to prove that it is inhabited if seen as a type, and thus to prove $\mathcal{T} \vdash \exists t. \, \varepsilon(t, |A|)$. Notwithstanding, this would probably be very inefficient, because the theorem prover would not be adapted to this particular theory. It should be better to use theorem provers based on superdeduction, such as the proof assistant Lemuridæ which has to be adapted for intuitionistic logic. (For the moment, it can be used because classical super rules subsumes intuitionistic ones, but we cannot guarantee the conservativeness.) Possibly, we could also use tools based on deduction modulo—there exists translations between supernatural deduction and deduction modulo that relatively preserve proof structures [7]—using procedures such as ENAR [15] or TaMed [6]. But to fully be able to check proofs from assistants, we also need to encode inductive types, that are present in many provers, such as e.g. Coq. This can also naturally be done in superdeduction, as shown by a recent work of Allali and Brauner [2]. Another benefit of having such a translation is the simplicity of first-order logic, and the fact that it is well-known and widely used. First, the implementation of a first-order system is often easier than of an higher-order one, because it is closer to the machine representation. Indeed, the proof checking code of Lemuridæ [8] is very simple, especially compared to the type checker of Coq, which is but claimed to be relatively small. Second, people are often most used to work using first-order logic, even if they sometimes need higher-order constructs, and the Π constructs in the PTS may at first be disturbing. This paper also permits to make proof assistants cooperate, in the sense of Europa and Fellowship. Nevertheless, the translation of a library of one assistant into another will be its encoding using ε, which is probably not convenient to develop proofs in the other assistant. An idea is then to decode this, using for instance $||\varepsilon(t, \dot{\pi}_{\langle s_1, s_2, s_3 \rangle}(a, b))|| = \Pi x : ||\varepsilon(x, a)|| \, ||\varepsilon(tx, b)||$. However, such an naive decoding is not correct if we translate one PTS into another. It remains to be investigated in which cases the decoding is safe.

An important remaining issue is the decidability of type checking/proof search. Type checking is undecidable for PTS in general, but it is decidable for functional PTS that weakly normalizes [5]. In that case, the conservativeness proves that proof search in NJ^+ is at most semi-decidable, even if we are working modulo an undecidable congruence (λ_W). (Note that proof search for atomic sequents with well-formed context in NJ^+ corresponds to type checking in the PTS.) We conjecture that it is in fact decidable for atomic sequents, due to the strong correspondence between the two systems. In a related topic, we also have to look at the normalization of the NJ^+ system for PTS, i.e. to check if proof terms associated with a NJ^+ proof (which are in fact ρ-terms [10]) strongly normalize. Due to the conservativeness, there is a correspondence between a NJ^+ proof and a term of the PTS, at least when contexts are well formed. Abstraction corresponds to the introduction rule $(3)_I$ and Application to the elimination rule $(3)_{E2}$, so that β-redex of PTS are translated into ρ-redex of NJ^+. The normalization of this NJ^+ system is therefore probably equivalent to the normalization of the PTS. By the way, it remains to extend the rewrite systems with some rules to check for the well-formedness of contexts, to have a full correspondence.

We should also try to encode more deductive systems into superdeduction. Indeed, the naturalness of our translation into supernatural deduction probably comes from the fact that PTS are extensions of simply-typed λ-calculus which is isomorph to natural deduction. This could be seen as an attenuation of our claim that superdeduction is a powerful logical framework. It is but able to encode elaborated deduction systems and theories, such as PTS, as we have seen, HOL [14, 8], (higher-order) arithmetic [17, 9] and Zermelo's set theory [16].

References

[1] M. Abadi, L. Cardelli, P.-L. Curien, and J.-J. Lévy. Explicit substitutions. *Journal of Functional Programming*, 1(4):375–416, 1991.

[2] L. Allali and P. Brauner. A semantic normalization proof of systems with recursors. Submitted, 2008.

[3] F. Baader and T. Nipkow. *Term Rewriting and all That*. Cambridge University Press, 1998.

[4] H. Barendregt. *Handbook of logic in computer science*, volume 2, chapter Lambda calculi with types, pages 117–309. Oxford University Press, 1992.

[5] G. Barthe. The semi-full closure of pure type systems. In *Mathematical Foundations of Computer Science*, volume 1450 of *LNCS*, pages 316–325. Springer, 1998.

[6] R. Bonichon and O. Hermant. A semantic completeness proof for TaMed. In M. Hermann and A. Voronkov, editors, *LPAR*, volume 4246 of *LNCS*, pages 167–181. Springer, 2006.

[7] P. Brauner, G. Dowek, and B. Wack. Normalization in supernatural deduction and in deduction modulo. 2007.

[8] P. Brauner, C. Houtmann, and C. Kirchner. Principle of su-
perdeduction. In L. Ong, editor, *Proceedings of LICS*, pages
41–50, jul 2007.

[9] G. Burel. Unbounded proof-length speed-up in deduction
modulo. In J. Duparc and T. A. Henziger, editors, *CSL 2007*,
volume 4646 of *LNCS*, pages 496–511. Springer, 2007.

[10] H. Cirstea and C. Kirchner. The rewriting calculus — Part I
and II. *Logic Journal of the Interest Group in Pure and
Applied Logics*, 9:427–498, May 2001.

[11] D. Cousineau and G. Dowek. Embedding pure type sys-
tems in the lambda-pi-calculus modulo. In S. Ronchi
Della Rocca, editor, *TLCA*, volume 4583 of *LNCS*, pages
102–117. Springer, 2007.

[12] G. Dowek. Confluence as a cut elimination property. In
R. Nieuwenhuis, editor, *RTA*, volume 2706 of *LNCS*, pages
2–13. Springer, 2003.

[13] G. Dowek, T. Hardin, and C. Kirchner. Higher-order unifi-
cation via explicit substitutions. *Information and Computa-
tion*, 157(1/2):183–235, 2000.

[14] G. Dowek, T. Hardin, and C. Kirchner. HOL-λσ an inten-
tional first-order expression of higher-order logic. *Mathe-
matical Structures in Computer Science*, 11(1):1–25, 2001.

[15] G. Dowek, T. Hardin, and C. Kirchner. Theorem proving
modulo. *J. of Automated Reasoning*, 31(1):33–72, 2003.

[16] G. Dowek and A. Miquel. Cut elimination for zermelo's set
theory. Available on author's web page.

[17] G. Dowek and B. Werner. Arithmetic as a theory modulo. In
J. Giesl, editor, *RTA*, volume 3467 of *LNCS*, pages 423–437.
Springer, 2005.

[18] G. Gentzen. Untersuchungen über das logische Schliessen.
Mathematische Zeitschrift, 39:176–210, 405–431, 1934.
Translated in Szabo, editor., *The Collected Papers of Ger-
hard Gentzen* as "Investigations into Logical Deduction".

[19] H. Geuvers. The Church-Rosser property for beta-eta-
reduction in typed lambda-calculi. In *LICS*, pages 453–460.
IEEE Computer Society, 1992.

[20] F. Gutiérrez and B. C. Ruiz. Cut elimination in a class of
sequent calculi for pure type systems. *Electronic Notes in
Theoretical Computer Science*, 84, 2003.

[21] R. Harper, F. Honsell, and G. Plotkin. A framework for
defining logics. *Journal of the ACM*, 40(1):143–184, 1993.

[22] D. Kesner. Confluence of extensional and non-extensional
λ-calculi with explicit substitutions. *Theoretical Computer
Science*, 238(1–2):183–220, 2000.

[23] S. Lengrand, R. Dyckhoff, and J. McKinna. A sequent cal-
culus for Pure Type Systems. In Z. Esik, editor, *CSL*, volume
4207 of *LNCS*, pages 441–455. Springer, September 2006.

[24] F. Pfenning. The practice of logical frameworks. In *CAAP*,
volume 1059 of *LNCS*, pages 119–134. Springer, 1996.

[25] C. Sacerdoti Coen and F. Kirchner. *Fellowship*, 2006.
http://www.lix.polytechnique.fr/Labo/Florent.
Kirchner/fellowship/.

[26] A. Waaler and L. Wallen. *Handbook of Tableau Methods*,
chapter Tableaux for Intutionistic Logics, pages 255–296.
Kluwer Academic Publishers, Boston, 1999.

[27] B. Wack. Supernatural deduction. Manuscript, available at
http://www.loria.fr/~wack/papers/supernatural.
ps.gz, 2006.

Contents

A. Proofs of the equivalence between LB$^+$ and NJ$^+$

A.1. Properties of LB$^+$

Definition 7 (Proof skeleton). *The skeleton of a proof is the tree of the inference rules used in it.*

Definition 8 (Structural inclusion). *We say that a proof π is structurally included into a proof ϖ if the skeleton of π is included in the skeleton of ϖ in the sense of tree inclusion.*

Proposition 12 (Admissibility of the weakening). *If $\Gamma \vdash \Delta$ can be proved in LB$^+$, so does $\Gamma', \Gamma \vdash \Delta, \Delta'$, with the same skeleton.*

Proof. By induction on the proof and on the formulæ in the right-hand side of the rewrite system. □

Lemma 13. *Let R be the rewrite rule $A \to P$ with associated left super rule*

$$R_L \; \frac{(\Gamma, \Gamma_i \vdash \Delta_i, \Delta)_i}{\Gamma, A \vdash \Delta}$$

Suppose that for all i there exists some $C_i \in \Delta_i, \Delta$ such that $\Gamma, \Gamma_i \vdash C_i$ is provable.

Then there exists some $C \in \Delta$ such that either $\Gamma \vdash C$ is provable or $\Gamma, \Gamma_i \vdash \Delta_i, C$ is provable for all i. Note that in both cases, $\Gamma, A \vdash C$ is provable.

Proof. By induction on P we can easily show that right rules are of the form

$$R_L \; \frac{(\Gamma, \Gamma_i \vdash \Delta_i, \Delta)_{i \in I}}{\Gamma, A \vdash \Delta}$$

and that there exists $i_0 \in I$ such that for all $i \in I$, $\Gamma_{i_0} \subseteq \Gamma_i$.

We proceed by induction on P.

Suppose that for all i there exists some $C_i \in \Delta_i, \Delta$ such that $\Gamma, \Gamma_i \vdash C_i$ is provable.

If $C_{i_0} \in \Delta$, then $\Gamma, \Gamma_{i_0} \vdash C_{i_0}$ is provable, and by weakening, for all i we can prove $\Gamma, \Gamma_i \vdash \Delta_i, C_{i_0}$.

Suppose $C_{i_0} \in \Delta_{i_0}$. We can show that for all i, if some formula C is in $Delta_i$ then C appears on the left of some implication in P. Therefore $P = D[C_{i_0} \Rightarrow P']$ for some context D and proposition P'. Let S be the rewrite rule $B \to D[P']$. Then by definition, if $I' = \{i \in I : C_{i_0} \not\in \Delta_i\}$, the left rule for S is

$$S_L \; \frac{(\Gamma, \Gamma_i \vdash \Delta_i, \Delta)_{i \in I'}}{\Gamma, B \vdash \Delta}$$

By induction hypothesis, there exists some $C \in \Delta$ such that either $\Gamma \vdash C$ is provable, which permits to conclude, or for all $i \in I'$, $\Gamma, \Gamma_i \vdash \Delta_i, C$ is provable. It remains to be proved that for all $i \in I \setminus I'$, $\Gamma, \Gamma_i \vdash \Delta_i, C$, but if $i \in I \setminus I'$, then $\Gamma_{i_0} \subseteq \Gamma_i$ and $C_{i_0} \in \Delta_i$ so that we can conclude by weakening with the proof of $\Gamma, \Gamma_{i_0} \vdash C_{i_0}$. □

Note: The proof of $\Gamma, A \vdash C$ obtained at the end is structurally included in the proof of $\Gamma, A \vdash \Delta$ we get by applying R_L to the proofs of the supposition.

Lemma 14. *Let R be the rewrite rule $A \to P$.*
The associated right super rule is of the form

$$R_R \; \frac{(\Gamma, \Gamma_i \vdash A_i)_{i \in I} \qquad (\Gamma \vdash A_j, \Delta)_{j \in J}}{\Gamma \vdash A, \Delta}$$

Suppose for all $i \in I$ we can prove $\Gamma, \Gamma_i \vdash A_i$ and for all $j \in J$ there exists some $C_j \in A_j, \Delta$ such that we can prove $\Gamma \vdash C_j$.

Then either $\Gamma \vdash A_j$ is provable for all $j \in J$ or there exists some $C \in \Delta$ such that $\Gamma \vdash \Delta$. In both cases, there exists some $C \in A, \Delta$ such that $\Gamma \vdash C$ is provable.

Proof. To prove the form of the right rule, we proceed by induction on P. The only difficult case is the implication: suppose $P = P_1 \Rightarrow P_2$. Let S be the rewrite rule $B \to P_2$. By induction hypothesis, the right rule for S is of the form

$$S_R \frac{(\Gamma, \Gamma_i \vdash A_i)_{i \in I} \qquad (\Gamma \vdash A_j, \Delta)_{j \in J}}{\Gamma \vdash B, \Delta}$$

Then by definition the right rule for R is

$$S_R \frac{(\Gamma, P_1, \Gamma_i \vdash A_i)_{i \in I} \qquad (\Gamma, P_1 \vdash A_j)_{j \in J}}{\Gamma \vdash A, \Delta}$$

which is of the convenient form.

To prove the rest of the lemma, suppose for all $i \in I$ we can prove $\Gamma, \Gamma_i \vdash A_i$ and for all $j \in J$ there exists some $C_j \in A_j, \Delta$ such that we can prove $\Gamma \vdash C_j$. If there exists $j \in J$ such that $C_j \in \Delta$, then we conclude because $\Gamma \vdash C_j$. Otherwise, for all $j \in J$ we have $C_j = A_j$. That means that for all $j \in J$ we can prove $\Gamma \vdash A_j$ and by applying the right rule for R we get a proof of $\Gamma \vdash A$. $\qquad\square$

Note: The proof of $\Gamma \vdash C$ obtained at the end is structurally included in the proof of $\Gamma \vdash A, \Delta$ we get by applying R_R to the proofs of the supposition.

Proposition 15 (Constructivism). *If π is a proof of $\Gamma \vdash \Delta$ LB$^+$, then there exists some $C \in \Delta$ and of proof of $\Gamma \vdash C$ in LB$^+$ which is structurally included in π.*

Proof. By induction on the proof. If the last rule is an Axiom or a logical rule, we can easily conclude by induction hypothesis.

If it is a Cut

$$\text{Cut} \frac{\boxed{\pi} \quad \boxed{\varpi} \\ \Gamma, P \vdash \Delta \qquad \Gamma \vdash P, \Delta}{\Gamma \vdash \Delta}$$

by induction hypothesis on ϖ, there exists some $C \in P, \Delta$ such that there exists a proof ϖ' of $\Gamma \vdash C$ in LB$^+$. If C is in Δ, ϖ' is a convenient proof. If $C = P$, by weakening π' is also a proof of $\Gamma \vdash P, C$ By induction hypothesis on π, there exists some $C \in \Delta$ such that there exists a proof π' of $\Gamma, P \vdash C$ in LB$^+$. A convenient proof is therefore

$$\text{Cut} \frac{\boxed{\pi'} \quad \boxed{\varpi'} \\ \Gamma, P \vdash C \qquad \Gamma \vdash P, C}{\Gamma \vdash C}$$

If it is a left super rule, then we can conclude by induction hypothesis and Lemma 13, and if it is a right super rule, by induction hypothesis and Lemma 14. $\qquad\square$

Lemma 16. *If we have a formula P of degree d greater or equal than 2 (i.e. non-atomic), if we have proofs π of $\Gamma, P \vdash \Delta'$ and π' of $\Gamma' \vdash P, \Delta'$ of degree less than d, then we can a proof of $\Gamma, \Gamma' \vdash \Delta, \Delta'$ of degree less than d.*

Proof. First remark that if a proof is structurally included into another, then its degree is less or equal, and that weakening a proof preserve its degree.

First, let remark that we can consider that $\Delta' = \emptyset$. Indeed, by Proposition 15 with π', we can prove either $\Gamma' \vdash P$ or $\Gamma' \vdash \Delta'$ by a proof of degree less than d. In the second case we conclude by weakening.

By induction on π, π'.

1. If π ends with an axiom. Then either there is some $C \in \Gamma \cap \Delta$ in which case we can prove $\Gamma, \Gamma' \vdash \Delta$ with an axiom; or $P \in \Delta$, in which case π' is by weakening a proof of $\Gamma, \Gamma' \vdash \Delta$.

2. If π' ends with an axiom: dual to 1.

3. If π ends with a logical or super rule with the active formula $A \in \Gamma \cup \Delta$. If $A \in \Gamma$, let Γ'' be $\Gamma \setminus \{A\}$ and Δ'' be Δ, otherwise let Γ'' be Γ and Δ'' be $\Delta \setminus \{A\}$. π is

$$r \frac{\boxed{\pi_i} \qquad\qquad \boxed{\pi_j} \\ (\Gamma'', P, \Gamma_i \vdash \Delta_i, \Delta'')_i \qquad (\Gamma'', P, \Gamma_j \vdash \Delta_j)_j}{\Gamma, P \vdash \Delta}$$

By induction hypothesis, there exists proofs ϖ_i of degree less than d of $\Gamma'', \Gamma', \Gamma_i \vdash \Delta_i, \Delta''$ and proofs ϖ_j of $\Gamma'', \Gamma', \Gamma_j \vdash \Delta_j$.

We conclude with

$$r \frac{\boxed{\varpi_i} \qquad\qquad \boxed{\varpi_j} \\ (\Gamma'', \Gamma', \Gamma_i \vdash \Delta_i, \Delta'')_i \qquad (\Gamma'', \Gamma', \Gamma_j \vdash \Delta_j)_j}{\Gamma, \Gamma' \vdash \Delta}$$

4. If π' ends with a logical or super rule with the active formula $A \in \Gamma'$. Let Σ be $\Gamma' \setminus \{A\}$. π' is

$$r \frac{\boxed{\pi'_i} \\ (\Sigma, \Gamma_i \vdash \Delta_i, P)_i}{\Gamma' \vdash P}$$

By induction hypothesis, there exists proofs ϖ'_i of degree less than d of $\Gamma, \Sigma, \Gamma_i \vdash \Delta_i, \Delta$.

We conclude with

$$r \frac{\boxed{\varpi'_i} \\ (\Gamma, \Sigma, \Gamma_i \vdash \Delta_i, \Delta)_i}{\Gamma, \Gamma' \vdash \Delta}$$

5. If both π and π' end with logical or super rules with P as active formula. Because P is not atomic, these rules are in fact logical rules.

- If $P = Q \wedge R$, π is

$$\wedge_L \frac{\boxed{\pi_1} \quad \Gamma, Q, R \vdash \Delta}{\Gamma, Q \wedge R \vdash \Delta}$$

and π'

$$\wedge_R \frac{\boxed{\pi'_1} \quad \boxed{\pi'_2}}{\Gamma' \vdash Q \quad \Gamma \vdash R}{\Gamma' \vdash Q \wedge R}$$

We conclude with

$$\text{Cut} \frac{\text{Cut} \frac{\boxed{\pi_1} \quad \boxed{\pi'_2}}{\Gamma, \Gamma', Q, R \vdash \Delta \quad \Gamma, \Gamma', Q \vdash R, \Delta}}{\Gamma, \Gamma', Q \vdash \Delta} \quad \frac{\boxed{\pi'_1}}{\Gamma, \Gamma' \vdash Q, \Delta}}{\Gamma, \Gamma' \vdash \Delta}$$

- If P is $Q \Rightarrow R$, π is

$$\Rightarrow_L \frac{\boxed{\pi_1} \quad \boxed{\pi_2}}{\Gamma \vdash Q, \Delta \quad \Gamma, R \vdash \Delta}{\Gamma, Q \Rightarrow R \vdash \Delta}$$

and π' is

$$\Rightarrow_R \frac{\boxed{\pi'_1} \quad \Gamma', Q \vdash R}{\Gamma' \vdash Q \Rightarrow R}$$

We conclude with

$$\text{Cut} \frac{\text{Cut} \frac{\boxed{\pi_2} \quad \boxed{\pi'_1}}{\Gamma, \Gamma', Q, R \vdash \Delta \quad \Gamma, \Gamma', Q \vdash R, \Delta}}{\Gamma, \Gamma', Q \vdash \Delta} \quad \frac{\boxed{\pi_1}}{\Gamma, \Gamma' \vdash Q, \Delta}}{\Gamma, \Gamma' \vdash \Delta}$$

- If P is $\forall x. \ Q$, then π is

$$\forall_L \frac{\boxed{\pi_1} \quad \Gamma, \{t/x\}Q \vdash \Delta}{\Gamma, \forall x.Q \vdash \Delta}$$

for some t and π'

$$\forall_R \frac{\boxed{\pi'_1} \quad \Gamma' \vdash Q}{\Gamma' \vdash \forall x. \ Q} \ x \text{ not free in } \Gamma'$$

As x is free in Γ' we can replace it by t in π'_1 to get a proof ϖ'_1 of $\Gamma' \vdash \{t/x\}Q$ of the same degree.
We conclude with

$$\text{Cut} \frac{\boxed{\pi_1} \quad \boxed{\varpi'_1}}{\Gamma, , \Gamma'\{t/x\}Q \vdash \Delta \quad \Gamma, \Gamma' \vdash \{t/x\}Q, \Delta}{\Gamma, \Gamma' \vdash \Delta}$$

$\square$

Proposition 3 (Atomic LB$^+$). *If the rewrite system is ppf, and Γ and Δ contains only atomic formulæ, then $\Gamma \vdash \Delta$ is provable iff it is provable with only identity and super rules.*

Proof. First remark that the rewrite system is ppf iff the super rules of LB$^+$ decompose atomic formuæ into atomic formulæ.

Suppose there is a proof of $\Gamma \vdash \Delta$. By induction on this proof. As nor Γ neither Δ contain non atomic formulæ, there cannot be a logical rule at the root of the proof. If the rule at the root of the proof is an Axiom, we are done. If it is a super rule, we can conclude with the remark above by induction hypothesis. If it is a Cut, it remains to be proved that we only need Cuts around atomic formulæ.

Note that as Cut may not be admissible (see Figure 5), we cannot avoid having Cuts in our proofs. Nevertheless, we can use some kind of cut elimination procedure to reduce to the case of atomic Cut.

Let suppose the last rule is a Cut, we proceed by induction on the degree of P:

$$\text{Cut} \frac{\boxed{\pi} \quad \boxed{\varpi}}{\Gamma, P \vdash \Delta \quad \Gamma \vdash P, \Delta}{\Gamma \vdash \Delta}$$

If the degree d of P is greater than 1, we can apply the induction hypothesis on the subproof and Lemma 16 to get a proof of $\Gamma, \Gamma \vdash \Delta, \Delta$ of degree less than d. Because Γ and Δ contains only atomic formulæ, we can use Cut to simulate contraction:

$$\text{Cut} \frac{\Gamma, A, A \vdash \Delta \quad \text{Axiom} \frac{}{\Gamma, A \vdash A, \Delta}}{\Gamma, A \vdash \Delta}$$

The proof we obtain is therefore if degree less than the degree of P, so we can apply the induction hypothese to conclude.

$\square$

A.2. Translations from NJ$^+$ to LB$^+$

For convenience, proofs in NJ$^+$ in this section and the following will be represented *a la* Gentzen [18], instead of in sequent style.

Lemma 17. *Let R be the rewrite rule $A \to P$.*
An elimination rule for R is of the form

$$R_{Ep} \frac{A \quad H_1 \quad \cdots \quad H_n}{K}$$

14

A left rule for R is of the form

$$R_L \frac{(\Gamma,\Gamma_i \vdash \Delta_i,\Delta)_i}{\Gamma,A \vdash \Delta}$$

Suppose we have proofs of $\Gamma \vdash H_j$ in LB^+ for all j.
Then we can prove $\Gamma,\Gamma_i \vdash \Delta_i,K$ in LB^+(and thus also $\Gamma,A \vdash K$).

Proof. The form of the elimination and left rules can be shown by a simple induction.

The remaining can be proved by induction on the formula P.

- Suppose P is atomic or $P = \top$.

  Then the elimination rule of R is

  $$R_E \frac{A}{P}$$

  The left rule for R is

  $$R_L \frac{\Gamma,P \vdash \Delta}{\Gamma,A \vdash \Delta}$$

  We have

  $$\text{Axiom or } \top_R \frac{}{\Gamma,P \vdash P}$$

- Suppose P is $P_1 \Rightarrow P_2$.

  Let S be the rewrite rule $D \to P_2$.

  An elimination rule for S is of the form

  $$S_{Ep} \frac{D \quad H_1 \quad \cdots \quad H_n}{K}$$

  and elimination rules for R are then

  $$R_{Ep} \frac{A \quad P_1 \quad H_1 \quad \cdots \quad H_n}{K}$$

  The left rule for S is of the form

  $$S_L \frac{(\Gamma,\Gamma_i \vdash \Delta_i,\Delta)_i}{\Gamma,D \vdash \Delta}$$

  and then the left rule for R is

  $$R_L \frac{(\Gamma,\Gamma_i \vdash \Delta_i,\Delta)_i \quad \Gamma \vdash P_1,\Delta}{\Gamma,A \vdash \Delta}$$

Suppose we have proofs π_j of $\Gamma \vdash H_j$ and a proof π_0 of $\Gamma \vdash P_1$.

By induction hypothesis, there are proofs of $\Gamma,\Gamma_i \vdash \Delta_i,K$ for all i, and by weakening π_0 is a proof of $\Gamma \vdash P_1,K$.

Note: This could not be done in LJ^+, because we cannot have multiple formulæ in the right part of a sequent.

- Suppose P is $P_1 \wedge P_2$.

  Let R_1 be the rewrite rule $A_1 \to P_1$ and R_2 be the rewrite rule $A_2 \to P_2$.

  For $k \in \{1;2\}$, an elimination rule for R_k is of the form

  $$R_{kEp} \frac{A_k \quad H_{k,p}^1 \quad \cdots \quad H_{k,p}^n}{K_{k,p}}$$

  and elimination rules for R are then

  $$R_{Ek,p} \frac{A \quad H_{k,p}^1 \quad \cdots \quad H_{k,p}^n}{K_{k,p}}$$

  The left rule for R_k is of the form

  $$R_{kL} \frac{(\Gamma,\Gamma_i \vdash \Delta_i,\Delta)_{i \in I_k}}{\Gamma,A_k \vdash \Delta}$$

  and then the left rule for R is

  $$R_L \frac{(\Gamma,\Gamma_i,\Gamma_j \vdash \Delta_i,\Delta_j,\Delta)_{i \in I_1,j \in I_2}}{\Gamma,A \vdash \Delta}$$

  Let k be 1 or 2.

  Suppose we have proofs π_j of $\Gamma \vdash H_{k,p}^j$.

  By induction hypothesis with R_k, there are proofs of $\Gamma,\Gamma_i \vdash \Delta_i,K_{k,p}$ for all p and $i \in I_k$. By weakening, these are also proofs of $\Gamma,\Gamma_i,\Gamma_j \vdash \Delta_i,\Delta_j,K_{k,p}$ for all $j \in I_{3-k}$.

- Suppose P is $\forall x. Q$. Let t be a term actually substituted for x in R_L.

  Let S be the rewrite rule $D \to \{t/x\}Q$.

  An elimination rule for S is of the form

  $$S_{Ep} \frac{D \quad H_1 \quad \cdots \quad H_n}{K}$$

  and elimination rules for R are then

  $$R_{Ep} \frac{A \quad H_1 \quad \cdots \quad H_n}{K}$$

  The left rule for S is of the form

$$S_L \frac{(\Gamma, \Gamma_i \vdash \Delta_i, \Delta)_i}{\Gamma, \{t/x\}Q \vdash \Delta}$$

and then the left rule for R with t is

$$S_L \frac{(\Gamma, B, \Gamma_i \vdash \Delta_i, \Delta)_i}{\Gamma, A \vdash \Delta}$$

Suppose we have proofs π_j of $\Gamma \vdash H_j$.

By induction hypothesis, there are proofs of $\Gamma, \Gamma_i \vdash \Delta_i, K$ for all i.

$\square$

THEOREM 4.
A proof of C from Γ in NJ^+ can be translated into a proof of $\Gamma \vdash C$ in LB^+.

Proof. By induction on the NJ^+ proof:

- If the last inference rule is a logical rule, then we can use the same translation as [18].

- If the last rule is a super introduction rule, it is easy to see that super introduction rules for NJ^+ are subcases of super right rules for LB^+ with only one formula in the right part of the sequents. We can therefore conclude by induction.

- If the last rule is a super elimination rule of the form

$$R_{Ep} \frac{A \qquad H_1 \quad \cdots \quad H_n}{C}$$

By induction hypothesis, there is a LB^+ proof π_0 of $\Gamma \vdash A$ and there are LB^+ proofs π_j of $\Gamma \vdash H_j$ for all j.

By Lemma 17, there is a proof ϖ of $\Gamma, A \vdash C$ in LB^+.

By weakening, π_0 is also a proof of $\Gamma \vdash A, C$.

We can conclude with

$$\text{Cut} \frac{\boxed{\varpi} \qquad \boxed{\pi_0}}{\Gamma, A \vdash C \qquad \Gamma \vdash A, C}{\Gamma \vdash C}$$

$\square$

A.3. Translations from LB^+ to NJ^+

Lemma 18. *Suppose we have a LJ^+ proof whose last inference is*

$$R_L \frac{(\Gamma_i \vdash \Delta_i)_i}{\Gamma, A \vdash \Delta}$$

where R is $A \to P$. Suppose for all i there is some C_i in Δ_i such that we have supernatural deduction proofs π_i of C_i from assumptions Γ_i.

Then there is some C in Δ such that it is possible to build a supernatural deduction proof of C from assumptions Γ, A and using only elimination rules of R and the proofs π_i.

Proof. By induction on the formula P.

- Suppose P is atomic or $P = \top$.

  Then the last rule of the sequent calculus proof is in fact

  $$R_L \frac{\Gamma, P \vdash \Delta}{\Gamma, A \vdash \Delta}$$

  Therefore, if $C \in \Delta$ and π is a proof of C from Γ and P, then

  $$\Gamma \qquad R_E \frac{A}{P}$$
  $$\boxed{\pi}$$
  $$C$$

  is the proof we want.

- Suppose P is $P_1 \Rightarrow P_2$.

  Let S be the rewrite rule $D \to P_2$. By definition of the left rules, there exists some i_0 such that $(\Gamma_{i_0} \vdash \Delta_{i_0}) = (\Gamma \vdash P_1, \Delta)$ and

  $$S_L \frac{(\Gamma_i \vdash \Delta_i)_{i \neq i_0}}{\Gamma, D \vdash \Delta}$$

  Suppose that for all $i \neq i_0$ we have proofs π_i from Γ_i to C_i for some $C_i \in \Delta_i$. By induction hypothesis, we can build a proof ϖ of C from Γ, D, π_i for $i \neq i_0$, and elimination rules for S.

  An elimination rule for S is of the form

  $$S_{Ep} \frac{D \qquad H_1 \quad \cdots \quad H_n}{K}$$

  and elimination rules for R are then

  $$R_{Ep} \frac{A \qquad P_1 \qquad H_1 \quad \cdots \quad H_n}{K}$$

  By replacing S_{Ep} by R_{Ep} in ϖ we therefore obtain a proof of C from Γ, A, B, proofs π_i for $i \neq i_0$ and elimination rules for R.

  Suppose that there is some $C \in P_1, \Delta$ such that we have a proof π_{i_0} from Γ to C (or equivalently from Γ_{i_0} to $C_{i_0} \in \Delta_{i_0}$). There are two cases : Either C is in Δ, and thus π_{i_0} is a convenient proof. Or $C = P_1$, and then we can build a proof from Γ, A, proofs p_i for all i and elimination rules for R by replacing the assumption P_1 in ϖ by π_{i_0}.

- Suppose P is $P_1 \wedge P_2$.

  Let R_1 be the rewrite rule $A_1 \to P_1$ and R_2 be the rewrite rule $A_2 \to P_2$.

  The left rule for R_k is of the form

  $$R_{kL} \; \frac{(\Gamma, \Gamma_i \vdash \Delta_i, \Delta)_{i \in I_k}}{\Gamma, D \vdash \Delta}$$

  and then the left rule for R is

  $$R_L \; \frac{(\Gamma, \Gamma_i, \Gamma_j \vdash \Delta_i, \Delta_j, \Delta)_{i \in I_1, j \in I_2}}{\Gamma, A \vdash \Delta}$$

  Suppose that for all $i \in I_1, j \in I_2$ we have proofs $\pi_{i,j}$ from $\Gamma, \Gamma_i, \Gamma_j$ to C_i for some $C_i \in \Delta_i, \Delta_j, \Delta$. By induction hypothesis for R_1, there exists some $C_j \in \Delta_j, \Delta$ such that there is a proof ϖ_j of C_j from assumptions Γ, Γ_j, A_1, proof $\pi_{i,j}$ and elimination rules for R_1.

  An elimination rule for R_1 is of the form

  $$R_{1Ep} \; \frac{A_1 \quad H_1 \quad \cdots \quad H_n}{K}$$

  and some elimination rules for R are then

  $$R_{E1,p} \; \frac{A \quad H_1 \quad \cdots \quad H_n}{K}$$

  By replacing R_{1Ep} by $R_{E1,p}$ in ϖ_j we therefore obtain a proof ϖ'_j of C_j from Γ, Γ_j, A, proofs $\pi_{i,j}$ and elimination rules for R.

  By induction hypothesis for R_2 with these ϖ'_j, there exists some $C \in \Delta$ such that there is a proof ϖ of C from assumptions Γ, A_2, proof ϖ'_j and elimination rules for R_2. Once again, if we replace R_{2Ep} by $R_{E2,p}$ in ϖ we obtain a proof of C from Γ, A, proofs ϖ'_j and elimination rules for R. But proofs ϖ'_j use only proofs $\pi_{i,j}$ and elimination rules for R, so this proof is convenient.

- Suppose P is $\forall x. \, Q$. Let t be the term actually substituted for x in the super sequent calculus proof of $\Gamma, A \vdash \Delta$.

  Let S be the rewrite rule $D \to \{t/x\}Q$. By definition of the left rules, we have

  $$S_L \; \frac{(\Gamma_i \vdash \Delta_i)_i}{\Gamma, D \vdash \Delta}$$

  Suppose that for all i we have proofs π_i from Γ_i to C_i for some $C_i \in \Delta_i$. By induction hypothesis, there exists some $C \in \Delta$ such that we can build a proof ϖ of C from Γ, D, proofs from π_i and elimination rules for S.

  An elimination rule for S is of the form

$$S_{Ep} \; \frac{D \quad H_1 \quad \cdots \quad H_n}{K}$$

and elimination rules for R are then

$$R_{Ep} \; \frac{A \quad H_1 \quad \cdots \quad H_n}{K}$$

By replacing S_{Ep} by R_{Ep} in ϖ we therefore obtain a proof of C from Γ, A, proofs π_i and elimination rules for R.

$\square$

Lemma 19. *Suppose we have a* LB^+ *proof whose last inference is*

$$R_R \; \frac{(\Gamma_i \vdash \Delta_i)_i}{\Gamma \vdash A, \Delta}$$

where R is $A \to P$. Suppose for all i there is some C_i in Δ_i such that we have supernatural deduction proofs π_i of C_i from assumptions Γ_i.

Then there is some C in A, Δ such that it is possible to build a supernatural deduction proof of C from assumptions Γ and using only the proofs π_i, with the introduction rules of R at the root if $C = A$.

Proof. By induction on the formula P.

- Suppose $P = B$ is atomic.

  Then the last rule of the sequent calculus proof is in fact

  $$R_R \; \frac{\Gamma \vdash B, \Delta}{\Gamma \vdash A, \Delta}$$

  Suppose we have a proof π of B from Γ. Then

  $$\begin{array}{c} \Gamma \\ \boxed{\pi} \\ R_I \; \dfrac{B}{A} \end{array}$$

  is a convenient proof. Else, if we have a proof π of some other $C \in \Delta$ from Γ, this is a convenient proof.

- Suppose $P = \top$. Then the last rule of the sequent calculus proof is in fact

  $$R_R \; \frac{}{\Gamma \vdash A, \Delta}$$

  The convenient proof is therefore

  $$R_I \; \frac{}{\Gamma \vdash A}$$

- Suppose P is $P_1 \Rightarrow P_2$.

  Let S be the rewrite rule $D \to P_2$. By definition of the right rules,

  $$S_R \; \frac{(\Gamma_i \vdash \Delta_i)_i}{\Gamma, P_1 \vdash D}$$

  (Remark there is only D in the right part of the conclusion.)

  Suppose that for all i we have proofs π_i from Γ_i to C_i for some $C_i \in \Delta_i$. By induction hypothesis, we can build a proof ϖ of D from Γ, P_1, proofs π_i and the introduction rule for S.

  The introduction rule for S is of the form

  $$S_I \; \frac{\overset{[A_1^1, \cdots, A_1^{P_1}]}{B_1} \qquad \overset{[A_m^1, \cdots, A_m^{P_m}]}{B_m}}{D}$$

  and the introduction rule for R is then

  $$R_I \; \frac{\overset{[P_1, A_1^1, \cdots, A_1^{P_1}]}{B_1} \qquad \overset{[P_1, A_m^1, \cdots, A_m^{P_m}]}{B_m}}{A}$$

  By replacing S_I by R_I at the root of ϖ we therefore obtain a proof of A from Γ, proofs π_i and the introduction rule for R at the root. (P_1 is in the assumptions of all premises of R_I, so that it is no longer an assumption of the resulting proof.)

  Note: Here appears the difference between LB^+ and LK^+. If we allow multiple right formulæ in the premises of the right rule for $\Rightarrow$, we do not necessarily obtain a proof of D from Γ and P_1, but we can have a proof of some other $C \in \Delta$, and therefore we cannot conclude. This explains why, with the rewrite system $A \to (B \Rightarrow C) \land D$, the sequent $D \vdash A, B$ can be proved in LK^+ but not in LB^+ nor either A or B from D in NJ^+.

- Suppose P is $P_1 \land P_2$.

  Let R_1 be the rewrite rule $A_1 \to P_1$ and R_2 be the rewrite rule $A_2 \to P_2$.

  By definition of the right rules, there exists I_1 and I_2 such that for $k \in \{1; 2\}$

  $$R_{kR} \; \frac{(\Gamma_i \vdash \Delta_i)_{i \in I_k}}{\Gamma \vdash A_k, \Delta}$$

  and

  $$R_R \; \frac{(\Gamma_i \vdash \Delta_i)_{i \in I_1} \qquad (\Gamma_i \vdash \Delta_i)_{i \in I_2}}{\Gamma \vdash A, \Delta}$$

  Suppose that for all $i \in I_1 \cup I_2$ we have proofs π_i from Γ_i to C_i for some $C_i \in \Delta_i$. By induction hypothesis, there exists some $C_k \in A_k, \Delta$ such that we can build a proof ϖ_k of C from Γ, proofs π_i and the introduction rule for R_k.

  If $C_k \in \Delta$ for $k = 1$ or $k = 2$, ϖ_k is a convenient proof.

  In the other case, the introduction rule for R_k is of the form

  $$R_{kI} \; \frac{\overset{[A_{k,1}^1, \cdots, A_{k,1}^{P_1}]}{B_{k,1}} \qquad \overset{[A_{k,m}^1, \cdots, A_{k,m}^{P_m}]}{B_{k,m}}}{A_k}$$

  and the introduction rule for R is then

  $$R_I \; \frac{\left(\overset{[A_{k,j}^1, \cdots, A_{k,j}^{P_j}]}{B_{k,j}} \right)_{k,j}}{A}$$

  We can replace R_{1I} and R_{2I} at the root of ϖ_1 and ϖ_2 by R_I to obtain a proof of A from Γ, proofs π_i and the introduction rule for R at the root.

- Suppose P is $\forall x.\, Q$.

  Let S be the rewrite rule $D(x) \to Q$. Supposing x is free in Γ, by definition of the right rules,

  $$S_R \; \frac{(\Gamma_i \vdash \Delta_i)_i}{\Gamma \vdash D(x)}$$

  Suppose that for all i we have proofs π_i from Γ_i to C_i for some $C_i \in \Delta_i$. By induction hypothesis, we can build a proof ϖ of $D(x)$ from Γ, proofs π_i and the introduction rule for S.

  The introduction rule for S is of the form

  $$S_I \; \frac{\overset{[A_1^1, \cdots, A_1^{P_1}]}{B_1} \qquad \overset{[A_m^1, \cdots, A_m^{P_m}]}{B_m}}{D(x)}$$

  and the introduction rule for R is then

  $$R_I \; \frac{\overset{[A_1^1, \cdots, A_1^{P_1}]}{B_1} \qquad \overset{[A_m^1, \cdots, A_m^{P_m}]}{B_m}}{A} \quad x \text{ not free in the assumptions}$$

  But we are supposing that x is free in Γ, so that we can replace S_I by R_I in ϖ to get a proof of A from Γ, proofs π_i and the introduction rule for R.

$\square$

THEOREM 6.

A proof of the sequent $\Gamma \vdash C$ in LB^+ using only identity and super rules can be transformed into a proof in NJ^+ of C from Γ using only super rules.

Proof. We prove a stronger result: a proof of the sequent $\Gamma \vdash \Delta$ in LB^+ using only identity and super rules can be transformed into a proof in NJ^+ of C from Γ using only super rules for some $C \in \Delta$.

By induction on the LJ^+ proof:

- The last rule is an Axiom rule:

$$\text{Axiom} \ \frac{}{\Gamma, A \vdash A, \Delta}$$

Then we have a trivial supernatural deduction proof of A from assumption A.

- The last rule is a cut rule

$$\text{Cut} \ \frac{\Gamma, A \vdash \Delta \qquad \Gamma \vdash A, \Delta}{\Gamma \vdash \Delta}$$

By induction hypothesis on the right subproof there exists a proof π of either A or some $C \in \Delta$ from Γ using only super rules.

In the second case, π is a convenient proof. In the first case, by induction hypothesis on the left subproof, there exists a proof ϖ of some $C \in \Delta$ from Γ and A using only super rules. By replacing the assumptions A by π in ϖ, we obtain a proof of C from Γ.

- The last rule is a right super rule. We can conclude using Lemma 19 and induction hypothesis.

- The last rule is a left super rule. We can conclude using Lemma 18 and induction hypothesis.

$\square$

B. Interactions PTS terms and first-order terms

Lemma 20. *The translation is compatible with the α-conversion:*

$$|\lambda x : A \ B|_\ell = |\lambda y : A \ \{y/x\}B|_\ell$$

Proof. By simple induction. $\square$

Lemma 21. *If $x \notin \ell$,*

$$|A|_{\ell::x::nil} \left[\mathsf{lift}_{\overline{\ell}}(\mathsf{cons}(|u|_{\ell::nil}))\right] \overset{*}{\longleftrightarrow} |\{u/x\}A|_{\ell::nil} \ .$$

Proof. By induction on A.

- If $A = x$, then $|A|_{\ell::x::nil} \left[\mathsf{lift}_{\overline{\ell}}(\mathsf{cons}(|u|_{\ell::nil}))\right] = \underline{\overline{\ell}+1} \left[\mathsf{lift}_{\overline{\ell}}(\mathsf{cons}(|u|_{\ell::nil}))\right]$ and $|\{u/x\}A|_{\ell::nil} = |u|_{\ell::nil}$. We can conclude with [22, Corollary 3.2].

- If $A = y \in \ell$, then $|A|_{\ell::x::nil} \left[\mathsf{lift}_{\overline{\ell}}(\mathsf{cons}(|u|_{\ell::nil}))\right] = \underline{m} \left[\mathsf{lift}_{\overline{\ell}}(\mathsf{cons}(|u|_{\ell::nil}))\right]$ with $m < \overline{\ell} + 1$, and $|\{u/x\}A|_{\ell::nil} = |y|_{\ell::nil} = \underline{m}$. We can conclude with [22, Corollary 3.2].

- If $A = y \notin \ell :: x$, then $|A|_{\ell::x::nil} \left[\mathsf{lift}_{\overline{\ell}}(\mathsf{cons}(|u|_{\ell::nil}))\right] = y \left[\mathsf{shift}\right]^{\overline{\ell}+1} \left[\mathsf{lift}_{\overline{\ell}}(\mathsf{cons}(|u|_{\ell::nil}))\right]$, and $|\{u/x\}A|_{\ell::nil} = |y|_{\ell::nil} = y \left[\mathsf{shift}\right]^{\overline{\ell}}$. By [22, Corollary 5.7],

$$\begin{aligned} &y \left[\mathsf{shift}\right]^{\overline{\ell}+1} \left[\mathsf{lift}_{\overline{\ell}}(\mathsf{cons}(|u|_{\ell::nil}))\right] \\ &\overset{*}{\longleftrightarrow} \quad y \left[\mathsf{shift}\right] \left[\mathsf{cons}(|u|_{\ell::nil})\right] \left[\mathsf{shift}\right]^{\overline{\ell}} \\ &\overset{*}{\longleftrightarrow} \quad y \left[\mathsf{shift}\right]^{\overline{\ell}} \end{aligned}$$

- The interesting case is the abstraction over x, with $y \neq x$:

$$\begin{aligned} &|\lambda x : A \ B|_{\ell::x::nil} \left[\mathsf{lift}_{\overline{\ell}}(\mathsf{cons}(|u|_{\ell::nil}))\right] \\ &= \quad |\lambda y : A \ \{y/x\}B|_{\ell::x::nil} \left[\mathsf{lift}_{\overline{\ell}}(\mathsf{cons}(|u|_{\ell::nil}))\right] \\ &\overset{*}{\longleftrightarrow} \quad \lambda \left(|\{y/x\}B|_{y::\ell::x::nil} \left[\mathsf{lift}_{\overline{\ell}+1}(\mathsf{cons}(|u|_{\ell::nil}))\right]\right) \\ &\overset{*}{\underset{\text{IH}}{\longleftrightarrow}} \quad \lambda \left(|\{u/x\}\{y/x\}B|_{y::\ell::nil}\right) \\ &= \quad \lambda \left(|\{y/x\}B|_{y::\ell::nil}\right) \\ &= \quad |\lambda y : A \ \{y/x\}B|_{\ell::nil} \\ &= \quad |\lambda x : A \ B|_{\ell::nil} \\ &= \quad |\{u/x\}\lambda x : A \ B|_{\ell::nil} \end{aligned}$$

$\square$

Lemma 8. $|A|_{x::nil} \left[\mathsf{cons}(|u|)\right] \overset{*}{\longleftrightarrow} |\{u/x\}A|$.

Proof. Corollary of the previous lemma. $\square$

Lemma 8bis. *If $|A| \overset{*}{\longrightarrow} a$ then there exists A' such that $a \overset{*}{\longrightarrow} |A'|$.*

Proof. By definition, the W-normal form of a is a pure term [22, Definition 3.4], and it can therefore be translated back into some A' such that $a \overset{*}{\longrightarrow}_W |A'|$. $\square$

Lemma 22. *If z does not appear in ℓ, $|A|_{\ell::z::nil} = \{\underline{1}/z[\mathsf{shift}]\} \left(|A|_{\ell::nil} \left[\mathsf{lift}_{\overline{\ell}}(\mathsf{shift})\right]\right)$*

Proof. By induction on A.

- If $A = z$, then $|A|_{\ell::z::nil} = \overline{\ell}+1$ and $\{\underline{1}/z[\mathsf{shift}]\}\left(|A|_{\ell::nil}\left[\mathsf{lift}_{\overline{\ell}}(\mathsf{shift})\right]\right) = \{\underline{1}/z[\mathsf{shift}]\}\left(z[\mathsf{shift}]^{\overline{\ell}}\left[\mathsf{lift}_{\overline{\ell}}(\mathsf{shift})\right]\right)$ which is by [22, Corollary 5.5] $\{\underline{1}/z[\mathsf{shift}]\}\left(z[\mathsf{shift}]^{\overline{\ell}+1}\right) = \underline{1}[\mathsf{shift}]^{\overline{\ell}}$ which is by definition of the scheme $\overline{\ell}+1$.

- If $A = y \in \ell$, then $|A|_{\ell::z::nil} = \underline{m}$ with $m < \overline{\ell}$ and $\{\underline{1}/z[\mathsf{shift}]\}\left(|A|_{\ell::nil}\left[\mathsf{lift}_{\overline{\ell}}(\mathsf{shift})\right]\right) = \{\underline{1}/z[\mathsf{shift}]\}\left(\underline{m}\left[\mathsf{lift}_{\overline{\ell}}(\mathsf{shift})\right]\right)$ which is by [22, Corollary 3.2] $\{\underline{1}/z[\mathsf{shift}]\}\underline{m} = \underline{m}$.

- If $A = y \notin \ell :: x$, then $|A|_{\ell::z::nil} = y[\mathsf{shift}]^{\overline{\ell}+1}$ and $\{\underline{1}/z[\mathsf{shift}]\}\left(|A|_{\ell::nil}\left[\mathsf{lift}_{\overline{\ell}}(\mathsf{shift})\right]\right) = \{\underline{1}/z[\mathsf{shift}]\}\left(y[\mathsf{shift}]^{\overline{\ell}}\left[\mathsf{lift}_{\overline{\ell}}(\mathsf{shift})\right]\right)$ which is by [22, Corollary 5.5] $\{\underline{1}/z[\mathsf{shift}]\}\left(y[\mathsf{shift}]^{\overline{\ell}+1}\right) = y[\mathsf{shift}]^{\overline{\ell}+1}$.

- The interesting case is the abstraction over z, with $y \neq z$:

$$
\begin{aligned}
&|\lambda z : A\ B|_{\ell::z::nil}\\
&=_{\alpha} \quad |\lambda y : A\ \{y/z\}B|_{\ell::z::nil}\\
&= \quad \lambda\left(|\{y/z\}B|_{y::\ell::z::nil}\right)\\
&\overset{*}{\underset{\mathsf{IH}}{\longleftrightarrow}} \quad \lambda\left(\{\underline{1}/z[\mathsf{shift}]\}\left(|\{y/z\}B|_{y::\ell::nil}\left[\mathsf{lift}_{\overline{\ell}+1}(\mathsf{shift})\right]\right)\right)\\
&= \quad \{\underline{1}/z[\mathsf{shift}]\}\lambda\left(|\{y/z\}B|_{y::\ell::nil}\left[\mathsf{lift}_{\overline{\ell}+1}(\mathsf{shift})\right]\right)\\
&\longleftarrow \quad \{\underline{1}/z[\mathsf{shift}]\}\left(\lambda|\{y/z\}B|_{y::\ell::nil}\right)\left[\mathsf{lift}_{\overline{\ell}}(\mathsf{shift})\right]\\
&= \quad \{\underline{1}/z[\mathsf{shift}]\}\left(|\lambda y : A\ \{y/z\}B|_{\ell::nil}\right)\left[\mathsf{lift}_{\overline{\ell}}(\mathsf{shift})\right]\\
&=_{\alpha} \quad \{\underline{1}/z[\mathsf{shift}]\}\left(|\lambda z : A\ B|_{\ell::nil}\right)\left[\mathsf{lift}_{\overline{\ell}}(\mathsf{shift})\right]
\end{aligned}
$$

$\square$

Lemma 8ter. $|A|_{z::nil} = \{\underline{1}/z[\mathsf{shift}]\}|A|_{nil}[\mathsf{shift}]$

Proof. Corollary of the previous lemma $\qquad\square$

Lemma 8quater (Reduction under Π). • *If $\langle s_1, s_2, s_3\rangle \in \mathcal{R}$, $\Gamma \vdash_{\mathrm{PTS}} A : s_1$ and $\Gamma[x : A] \vdash_{\mathrm{PTS}} B : s_2$, x is not free in b, $a \overset{*}{\longrightarrow} |A|^{\Gamma}$ and $b[\mathsf{cons}(x)] \longrightarrow |B|^{\Gamma[x:A]}$ then $\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b) \overset{*}{\longleftrightarrow} |\Pi x : A\ B|^{\Gamma}$.*

- *If $\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b) \overset{*}{\longrightarrow} |C|$ then there exists A, B such that $C = \Pi x : A\ B$.*

Proof.

$$
\begin{aligned}
|\Pi x : A\ B| &= \quad \dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a, |B|_{x::nil})\\
&\overset{*}{\underset{\text{Lem. 8}}{\longleftrightarrow}} \quad \dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a, (\{\underline{1}/x[\mathsf{shift}]\}|B|_{nil}[\mathsf{shift}]))\\
&\overset{*}{\underset{\text{Lem. 8}}{\longleftarrow}} \quad \dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a, (\{\underline{1}/x[\mathsf{shift}]\}b[\mathsf{cons}(x)][\mathsf{shift}]))\\
&\overset{*}{\underset{\text{[22, Cor. 5.9]}}{\longleftarrow}} \quad \dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a, (\{\underline{1}/x[\mathsf{shift}]\}b[\mathsf{lift}(\mathsf{shift})][\mathsf{cons}(x[\mathsf{shift}])]))\\
&\underset{x \text{ not free in } b}{=} \quad \dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a, (b[\mathsf{lift}(\mathsf{shift})][\mathsf{cons}(\underline{1})]))\\
&\overset{*}{\underset{\text{[22, Lemma 4.5]}}{\longrightarrow}} \quad \dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a, b)
\end{aligned}
$$

$\square$

C. Why are several $\dot{\pi}$ necessary?

Because of rule $(2)_{E1}$: in the calculus of construction, we could get Girard's paradox:

$$
(2)_{E1}^{\langle *,*,*\rangle}\dfrac{(2)_{I}^{\langle \Box,*,*\rangle}\dfrac{(1)_{I}\dfrac{}{\vdash^{+}\varepsilon(\star,\Box)} \quad \text{Axiom}\dfrac{}{\varepsilon(z,\star)\vdash^{+}\varepsilon(z,\star)}}{\vdash^{+}\varepsilon(\dot{\pi}(\star,\underline{1}),\star)} \quad \underline{1}[\mathsf{cons}(z)]\overset{*}{\longrightarrow}z}{\vdash^{+}\varepsilon(\star,\star)}
$$