On the Strong Normalisation of Intuitionistic Natural Deduction with Permutation-Conversions

Philippe de Groote

LORIA UMR n^o 7503 – INRIA
Campus Scientifique, B.P. 239
54506 Vandœuvre lès Nancy Cedex – France
e-mail: Philippe.de.Groote@loria.fr

Abstract. We present a modular proof of the strong normalisation of intuitionistic logic with permutation-conversions. This proof is based on the notions of negative translation and CPS-simulation.

1 Introduction

Natural deduction systems provide a notion of proof that is more compact (or, quoting Girard [6], more *primitive*) than that of sequent calculi. In particular, natural deduction is better adapted to the study of proof-normalisation procedures. This is true, at least, for the intuitionistic systems, where proof-normalisation expresses the computational content of the logic. Nevertheless, even in the intuitionistic case, the treatments of disjunction and existential quantification are problematic. This is due to the fact that the elimination rules of these connectives introduce arbitrary formulas as their conclusions. Consequently, in order to satisfy the subformula property, the so-called permutation-conversions are needed.

Strong normalisation proofs for intuitionistic logic [12] are more intricate in the presence of permutation-conversions. For instance, the proofs given in textbooks such as [6] and [14] do not take permutation-conversions into account.[1] In this paper, we revisit this problem and present a simple proof of the strong normalisation of intuitionistic logic with permutation-conversions. This proof, which is inspired by a similar proof in [4], has several advantages:

- It is modular and, therefore, easily adaptable to other systems. Indeed, the problem related to the interaction between permutation- and detour-conversions is avoided (see Lemma 7, in Section 5).
- It is based on a continuation-passing-style interpretation of intuitionistic logic, which sheds light on the computational content of the several conversion rules. In particular, it shows that the computational content of permutation-conversions is nil.

[1] In [6], extending the proof to permutation-conversions is left to the reader while in [14], the technique that is used is not adapted to the case of permutation-conversions.

– It is based on an arithmetisable translation of intuitionistic logic into the simply typed λ-calculus. Consequently, when combined with an arithmetisable proof of strong normalisation of the simply typed λ-calculus (see [3], for instance), it yields a completely arithmetisable proof of the strong normalisation of intuitionistic logic. This must be contrasted with the proof in [14], which is based on an interpretation into higher-order Heyting arithmetic.

The paper is organised as follows.

Section 2 is an introduction to the proof-theory of intuitionistic positive propositional logic (**IPPL**). In particular, we define the notions of detour- and permutation-conversions by means of an associated λ-calculus ($\lambda^{\to\wedge\vee}$). Our presentation is essentially inspired by [14].

In Section 3, we establish the strong normalisation of **IPPL** with respect to permutation-conversions. The proof consists simply in assigning a norm to the untyped terms of $\lambda^{\to\wedge\vee}$, and showing that this norm strictly decreases by permutation conversion.

Section 4 provides a negative translation of **IPPL** into the implicative fragment of intuitionistic logic. At the level of the proofs, this negative translation corresponds to a CPS-simulation of $\lambda^{\to\wedge\vee}$ into the simply typed λ-calculus. We show that this CPS-simulation (slightly modified) commutes with the relations of detour-conversion and β-reduction, from which we conclude that $\lambda^{\to\wedge\vee}$ is strongly normalisable with respect to detour-conversions.

In Section 5, we collect the results of the two previous sections in order to show that $\lambda^{\to\wedge\vee}$ is strongly normalisable with respect to detour- and permutation-conversions mixed together. To this end, we show that the modified CPS-translation of Section 4 interprets the relation of permutation-conversion as equality. This means that we have found a negative translation commuting with the permutation-conversions, answering a problem raised by Mints [9].

In Section 6 and 7, we extend our proof, respectively, to positive first-order intuitionistic logic (by adding quantifiers), and then to full first-order intuitionistic logic (by adding negation).

2 Intuitionistic positive propositional logic

2.1 Natural deduction

The formulas of intuitionistic positive propositional logic (**IPPL**) are built up from an alphabet of atomic propositions $\mathcal{A}$ and the connectives $\to$, $\wedge$, and $\vee$ according to the following grammar:

$$\mathcal{F} \ ::= \ \mathcal{A} \mid \mathcal{F} \to \mathcal{F} \mid \mathcal{F} \wedge \mathcal{F} \mid \mathcal{F} \vee \mathcal{F}$$

Following Gentzen [5], the meaning of the connectives is specified by the introduction and elimination rules of the following natural deduction system (where a bracketed formula corresponds to a hypothesis that may be discarded when applying the rule).

$$\frac{\begin{array}{c}[\,\alpha\,]\\ \beta\end{array}}{\alpha \to \beta} \qquad \frac{\alpha \to \beta \quad \alpha}{\beta}$$

$$\frac{\alpha \quad \beta}{\alpha \wedge \beta} \qquad \frac{\alpha \wedge \beta}{\alpha} \qquad \frac{\alpha \wedge \beta}{\beta}$$

$$\frac{\alpha}{\alpha \vee \beta} \qquad \frac{\beta}{\alpha \vee \beta} \qquad \frac{\alpha \vee \beta \quad \begin{array}{c}[\,\alpha\,]\\ \gamma\end{array} \quad \begin{array}{c}[\,\beta\,]\\ \gamma\end{array}}{\gamma}$$

As observed by Prawitz [11], an introduction immediately followed by an elimination corresponds to a *detour* that can be (locally) eliminated. Consider, for instance, the case of implication:

$$\dfrac{\dfrac{\begin{array}{c}[\,\alpha\,]\\ \vdots\ \Pi_1\\ \beta\end{array}}{\alpha \to \beta}\,(\text{Intro.}) \qquad \begin{array}{c}\vdots\ \Pi_2\\ \alpha\end{array}}{\beta}\,(\text{Elim.})$$

Π_1 is a proof of β under the hypothesis α. On the other hand, Π_2 is a proof of α. Consequently, one may obtain a direct proof of β by grafting Π_2 at every place where α occurs in Π_1 as a hypothesis discarded by Rule (Elim.):

$$\begin{array}{c}\vdots\ \Pi_2\\ \alpha\\ \vdots\ \Pi_1\\ \beta\end{array}$$

When such local reduction steps allow any proof to be transformed into a proof without detour, one says that the given natural deduction system satisfies the *normalisation property*. Moreover, when this property holds independently of the strategy that is used in applying the reduction steps, one says that the system satisfies the *strong normalisation property*.

2.2 Natural deduction as a term calculus

As well-known, there exists a correspondence between natural deduction systems and typed λ-calculi, namely, the Curry-Howard isomorphism [2, 7]. This correspondence, which is described in the table below, allows natural deduction proofs to be denoted by terms.

Natural deduction	λ-calculus
propositions	types
connectives	type constructors
proofs	terms
introduction rules	term constructors
elimination rules	term destructors
active hypothesis	free variables
discarded hypothesis	bound variables

In the case of **IPPL**, the corresponding term calculus is the simply typed λ-calculus with product and coproduct, which we will call $\lambda^{\to\wedge\vee}$. In particular, introduction and elimination rules for conjunction correspond to pairing and projection functions, while introduction and elimination rules for disjunction correspond to injection and case analysis functions. The syntax is described by the following grammar, where $\mathcal{X}$ is a set of variables:

$$\mathcal{T} ::= \mathcal{X} \mid \lambda\mathcal{X}.\,\mathcal{T} \mid (\mathcal{T}\,\mathcal{T}) \mid \mathbf{p}(\mathcal{T},\mathcal{T}) \mid \mathbf{p}_1\,\mathcal{T} \mid \mathbf{p}_2\,\mathcal{T} \mid \mathbf{k}_1\,\mathcal{T} \mid \mathbf{k}_2\,\mathcal{T} \mid \mathbf{D}_{\mathcal{X},\mathcal{X}}(\mathcal{T},\mathcal{T},\mathcal{T})$$

and the typing rules are as follows:

$$\frac{\begin{array}{c}[\,x:\alpha\,]\\ M:\beta\end{array}}{\lambda x.\,M:\alpha\to\beta} \qquad \frac{M:\alpha\to\beta \quad N:\alpha}{M\,N:\beta}$$

$$\frac{M:\alpha \quad N:\beta}{\mathbf{p}(M,N):\alpha\wedge\beta} \qquad \frac{M:\alpha\wedge\beta}{\mathbf{p}_1\,M:\alpha} \qquad \frac{M:\alpha\wedge\beta}{\mathbf{p}_2\,M:\beta}$$

$$\frac{M:\alpha}{\mathbf{k}_1\,M:\alpha\vee\beta} \qquad \frac{M:\beta}{\mathbf{k}_2\,M:\alpha\vee\beta} \qquad \frac{M:\alpha\vee\beta \quad \begin{array}{c}[\,x:\alpha\,]\\ N:\gamma\end{array} \quad \begin{array}{c}[\,y:\beta\,]\\ O:\gamma\end{array}}{\mathbf{D}_{x,y}(M,N,O):\gamma}$$

2.3 Detour-conversion rules

The Curry-Howard isomorphism is not only a simple matter of notation. Its deep meaning lies in the relation existing between proof normalisation and λ-term evaluation. Indeed an introduction immediately followed by an elimination corresponds to a term destructor applied to term constructor. Consequently, Prawitz's detour elimination steps amount to evaluation steps. For instance, the detour elimination step for implication corresponds exactly to the familiar notion of β-reduction:

$$\frac{\dfrac{\begin{array}{c}[\,x:\alpha\,]\\ \vdots\ \Pi_1\\ M:\beta\end{array}}{\lambda x.\,M:\alpha\to\beta} \qquad \begin{array}{c}\vdots\ \Pi_2\\ N:\alpha\end{array}}{(\lambda x.\,M)\,N:\beta} \quad\longrightarrow\quad \begin{array}{c}\vdots\ \Pi_2\\ N:\alpha\\ \vdots\ \Pi_1\\ M[x{:=}N]:\beta\end{array}$$

It is therefore possible to specify the detour elimination steps as simple rewriting rules between λ-terms. Following [14], these rules are called *detour-conversion* rules. We use "$\to_D$" to denote the corresponding one-step reduction relation between λ-terms and, following [1], we use "$\xrightarrow{+}_D$" and "$\twoheadrightarrow_D$" to denote, repectively, the transitive closure and the transitive, reflexive closure of "$\to_D$".

Definition 1. *(Detour-conversions of $\lambda^{\to\wedge\vee}$)*

1. $(\lambda x.\, M)\, N \to_D M[x{:=}N]$
2. $\mathbf{p}_1\, \mathbf{p}(M, N) \to_D M$
3. $\mathbf{p}_2\, \mathbf{p}(M, N) \to_D N$
4. $\mathbf{D}_{x,y}(\mathbf{k}_1\, M, N, O) \to_D N[x{:=}M]$
5. $\mathbf{D}_{x,y}(\mathbf{k}_2\, M, N, O) \to_D O[y{:=}M]$ ∎

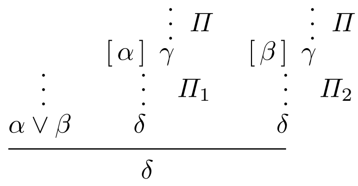
It must be clear that the Curry-Howard isomorphism allows one to reduce proof normalisation problems to λ-term normalisation problems. In particular, the strong normalisation of intuitionistic implicative logic corresponds to the strong normalisation of the simply typed λ-calculus [1] (which is, by the way, the only normalisation result we assume in this paper). There is, however, a slight difference that must be stressed. The grammar given in Section 2.2 defines untyped λ-terms, some of which do not correspond to natural deduction proofs. Consequently, the question whether the untyped λ-terms satisfy some normalisation property has no direct equivalent in the logical setting.

2.4 Permutation-conversion rules

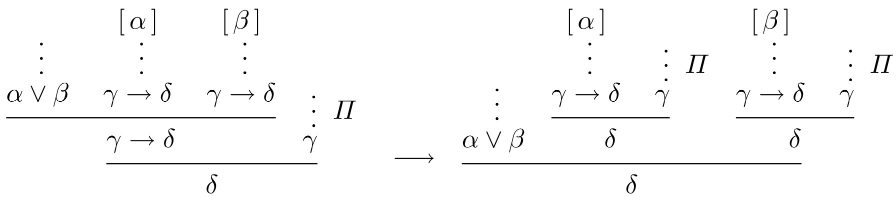
In the disjunction free fragment of **IPPL**, the normal proofs (i.e., the proofs without detour) satisfy the subformula property. This means that if Π is a normal proof of a formula α under a set of hypotheses Γ, then each formula occurring in Π is a subformula of a formula in $\{\alpha\} \cup \Gamma$ [11]. In the presence of disjunction, the detour-conversions of Definition 1 are no longer sufficient to guarantee the subformula property. Consider the following example:

$$\dfrac{\dfrac{\begin{matrix}\vdots\\ \alpha\vee\beta\end{matrix}\qquad \dfrac{\begin{matrix}[\alpha,\gamma]\\ \vdots\ \Pi_1\\ \delta\end{matrix}}{\gamma\to\delta}\,(\text{Intro.})\qquad \dfrac{\begin{matrix}[\beta,\gamma]\\ \vdots\ \Pi_2\\ \delta\end{matrix}}{\gamma\to\delta}\,(\text{Intro.})}{\gamma\to\delta}\qquad \begin{matrix}\vdots\ \Pi\\ \gamma\end{matrix}}{\delta}\,(\text{Elim.})$$

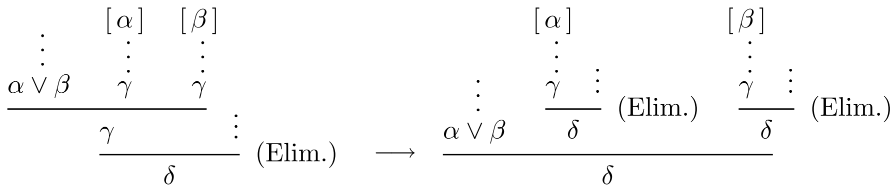
A priori there is no reason why γ and $\gamma \to \delta$ would be subformulas of δ or of any hypothesis from which δ is derived. This is due to the fact that there are introduction rules followed by an elimination rule. Indeed, one would like to reduce the above example as follows:



However, such a reduction is not possible by applying only the detour-conversion rules of Definition 1 because, in the above example, the elimination rule does not *immediately* follow the introduction rules. For this reason, some other conversion rules are needed, the so-called *permutation-conversion.* For instance, to reduce the above example, we need the following rule:



The above conversion, which concerns the implication elimination rule, obeys a general scheme:



All the permutation-conversions may be obtained, from the above scheme, by replacing Rule (Elim.) by the different elimination rules of **IPPL**. Of course, it is also possible to express these permutation-conversions as rewriting rules between λ-terms. This is achieved in the following definition.

Definition 2. *(Permutation-conversions of $\lambda^{\rightarrow\wedge\vee}$)*

1. $\mathbf{D}_{x,y}(M, N, O)\, P \rightarrow_P \mathbf{D}_{x,y}(M, N\, P, O\, P)$
2. $\mathbf{p}_1\, \mathbf{D}_{x,y}(M, N, O) \rightarrow_P \mathbf{D}_{x,y}(M, \mathbf{p}_1\, N, \mathbf{p}_1\, O)$
3. $\mathbf{p}_2\, \mathbf{D}_{x,y}(M, N, O) \rightarrow_P \mathbf{D}_{x,y}(M, \mathbf{p}_2\, N, \mathbf{p}_2\, O)$
4. $\mathbf{D}_{u,v}(\mathbf{D}_{x,y}(M, N, O), P, Q) \rightarrow_P \mathbf{D}_{x,y}(M, \mathbf{D}_{u,v}(N, P, Q), \mathbf{D}_{u,v}(O, P, Q))$ ■

We are now in a position to state precisely the question addressed by the present paper: how can we give a modular proof of the strong normalisation of **IPPL** with respect to both the detour- and permutation-conversions? or, equivalently, how can we prove that the typed λ-terms of Section 2.2 satisfy the strong normalisation property with respect to the reduction relation induced by the union of the rewriting systems of definitions 1 and 2?

3 Strong Normalisation of permutation-conversions

In this section, we establish the strong normalisation of $\lambda^{\to\wedge\vee}$ with respect to permutation-conversions. The proof consists simply in assigning a norm to the λ-terms and then in proving that this norm (which we call the permutation degree) is decreasing under the reduction relation $\to_P$.

Definition 3. *(Permutation degree for $\lambda^{\to\wedge\vee}$)*

1. $|x| = 1$
2. $|\lambda x.\, M| = |M|$
3. $|M\, N| = |M| + \#M \times |N|$
4. $|\mathbf{p}(M, N)| = |M| + |N|$
5. $|\mathbf{p}_1\, M| = |M| + \#M$
6. $|\mathbf{p}_2\, M| = |M| + \#M$
7. $|\mathbf{k}_1\, M| = |M|$
8. $|\mathbf{k}_2\, M| = |M|$
9. $|\mathbf{D}_{x,y}(M, N, O)| = |M| + \#M \times (|N| + |O|)$

where:

10. $\#x = 1$
11. $\#\lambda x.\, M = 1$
12. $\#M\, N = \#M$
13. $\#\mathbf{p}(M, N) = 1$
14. $\#\mathbf{p}_1\, M = \#M$
15. $\#\mathbf{p}_2\, M = \#M$
16. $\#\mathbf{k}_1\, M = 1$
17. $\#\mathbf{k}_2\, M = 1$
18. $\#\mathbf{D}_{x,y}(M, N, O) = 2 \times \#M \times (\#N + \#O)$ ■

Lemma 1. *Let M and N be two λ-terms of $\lambda^{\to\wedge\vee}$ such that $M \to_P N$. Then $\#M = \#N$.*

Proof. Let $C[\,]$ be any context, i.e., a λ-term with a hole. It is straightforward that $\#C[M] = \#C[N]$ whenever $\#M = \#N$. Hence it remains to show that $\#$ is invariant under each rewriting rule of Definition 2.

$$\begin{aligned}\#\mathbf{D}_{x,y}(M, N, O)\, P &= \#\mathbf{D}_{x,y}(M, N, O)\\ &= 2 \times \#M \times (\#N + \#O)\\ &= 2 \times \#M \times (\#N\, P + \#O\, P)\\ &= \#\mathbf{D}_{x,y}(M, N\, P, O\, P)\end{aligned}$$

$$\begin{aligned}\#\mathbf{p}_i\, \mathbf{D}_{x,y}(M, N, O) &= \#\mathbf{D}_{x,y}(M, N, O)\\ &= 2 \times \#M \times (\#N + \#O)\\ &= 2 \times \#M \times (\#\mathbf{p}_i\, N + \#\mathbf{p}_i\, O)\\ &= \#\mathbf{D}_{x,y}(M, \mathbf{p}_i\, N, \mathbf{p}_i\, O)\end{aligned}$$

$$
\begin{aligned}
&\#\mathbf{D}_{u,v}(\mathbf{D}_{x,y}(M,N,O),P,Q)\\
&\qquad= 2 \times \#\mathbf{D}_{x,y}(M,N,O) \times (\#P + \#Q)\\
&\qquad= 4 \times \#M \times (\#N + \#O) \times (\#P + \#Q)\\
&\qquad= 2 \times \#M \times (2 \times \#N \times (\#P + \#Q) + 2 \times \#O \times (\#P + \#Q))\\
&\qquad= 2 \times \#M \times (\#\mathbf{D}_{u,v}(N,P,Q) + \#\mathbf{D}_{u,v}(O,P,Q))\\
&\qquad= \#\mathbf{D}_{x,y}(M,\mathbf{D}_{u,v}(N,P,Q),\mathbf{D}_{u,v}(O,P,Q))
\end{aligned}
$$

□

Lemma 2. *Let M and N be two λ-terms of $\lambda^{\to\wedge\vee}$ such that $M \to_P N$. Then $|M| > |N|$.*

Proof. *The proof is similar to that of the previous lemma. We show that the permutation degree is strictly decreasing under the rewriting rules of Definition 2.*

$$
\begin{aligned}
&|\mathbf{D}_{x,y}(M,N,O)\,P|\\
&\qquad= |\mathbf{D}_{x,y}(M,N,O)| + \#\mathbf{D}_{x,y}(M,N,O) \times |P|\\
&\qquad= |M| + \#M \times (|N| + |O|) + 2 \times \#M \times (\#N + \#O) \times |P|\\
&\qquad> |M| + \#M \times (|N| + |O|) + \#M \times (\#N + \#O) \times |P|\\
&\qquad= |M| + \#M \times (|N| + \#N \times |P| + |O| + \#O \times |P|)\\
&\qquad= |M| + \#M \times (|N\,P| + |O\,P|)\\
&\qquad= |\mathbf{D}_{x,y}(M,N\,P,O\,P)|
\end{aligned}
$$

$$
\begin{aligned}
|\mathbf{p}_i\,\mathbf{D}_{x,y}(M,N,O)| &= |\mathbf{D}_{x,y}(M,N,O)| + \#\mathbf{D}_{x,y}(M,N,O)\\
&= |M| + \#M \times (|N| + |O|) + 2 \times \#M \times (\#N + \#O)\\
&> |M| + \#M \times (|N| + |O|) + \#M \times (\#N + \#O)\\
&= |M| + \#M \times (|N| + \#N + |O| + \#O)\\
&= |M| + \#M \times (|\mathbf{p}_i\,N| + |\mathbf{p}_i\,O|)\\
&= |\mathbf{D}_{x,y}(M,\mathbf{p}_i\,N,\mathbf{p}_i\,O)|
\end{aligned}
$$

$$
\begin{aligned}
&|\mathbf{D}_{u,v}(\mathbf{D}_{x,y}(M,N,O),P,Q)|\\
&\qquad= |\mathbf{D}_{x,y}(M,N,O)| + \#\mathbf{D}_{x,y}(M,N,O) \times (|P| + |Q|)\\
&\qquad= |M| + \#M \times (|N| + |O|) + 2 \times \#M \times (\#N + \#O) \times (|P| + |Q|)\\
&\qquad> |M| + \#M \times (|N| + |O|) + \#M \times (\#N + \#O) \times (|P| + |Q|)\\
&\qquad= |M| + \#M \times (|N| + \#N \times (|P| + |Q|) + |O| + \#O \times (|P| + |Q|))\\
&\qquad= |M| + \#M \times (|\mathbf{D}_{u,v}(N,P,Q)| + |\mathbf{D}_{u,v}(O,P,Q)|)\\
&\qquad= |\mathbf{D}_{x,y}(M,\mathbf{D}_{u,v}(N,P,Q),\mathbf{D}_{u,v}(O,P,Q))|
\end{aligned}
$$

□

We immediately obtain the expected strong normalisation result from the above lemma.

Proposition 1. $\lambda^{\to\wedge\vee}$ *is strongly normalisable with respect to permutation-conversions.* □

Remark that this proposition also holds for the untyped terms. This fact confirms that the permutation-conversions do not have a real computational meaning. The fact that they are needed to obtain the subformula property may be seen as a defect of the syntax.

4 Negative translation and CPS-simulation

We now establish the strong normalisation of $\lambda^{\rightarrow\wedge\vee}$ with respect to detour-conversions. To this end we interpret **IPPL** into intuitionistic implicative logic by means of a negative translation. This corresponds to a translation of $\lambda^{\rightarrow\wedge\vee}$ into the simply typed λ-calculus. This translation must satisfy two requirements. On the one hand, it must provide a simulation of the detour-conversions. On the other hand it must be compatible with the permutation-conversions in a sense that will be explained. In order to satisfy the first requirement, the negative translation we use is a generalisation of the one used by Meyer and Wand in the implicative case [8], i.e., a generalisation of the translation induced by Plotkin's call-by-name CPS-translation [10].

Definition 4. *(Negative translation of* **IPPL***) The negative translation $\overline{\alpha}$ of any formula α of* **IPPL** *is defined as:*

$$\overline{\alpha} = \sim\sim\alpha^{\circ}$$

where

$$\sim\alpha = \alpha \rightarrow o$$

for some distinguished atomic proposition o (that is not used elsewhere), and where:

1. $a^{\circ} = a$
2. $(\alpha \rightarrow \beta)^{\circ} = \overline{\alpha} \rightarrow \overline{\beta}$
3. $(\alpha \wedge \beta)^{\circ} = \sim(\overline{\alpha} \rightarrow \sim\overline{\beta})$
4. $(\alpha \vee \beta)^{\circ} = \sim\overline{\alpha} \rightarrow \sim\sim\overline{\beta}$ ■

Then we accomodate Plotkin's call-by-name simulation to the case of $\lambda^{\rightarrow\wedge\vee}$.

Definition 5. *(CPS-translation of $\lambda^{\rightarrow\wedge\vee}$)*

1. $\overline{x} = \lambda k.\, x\, k$
2. $\overline{\lambda x.\, M} = \lambda k.\, k\,(\lambda x.\, \overline{M})$
3. $\overline{(M\, N)} = \lambda k.\, \overline{M}\,(\lambda m.\, m\, \overline{N}\, k)$
4. $\overline{\mathbf{p}(M, N)} = \lambda k.\, k\,(\lambda p.\, p\, \overline{M}\, \overline{N})$
5. $\overline{\mathbf{p}_1\, M} = \lambda k.\, \overline{M}\,(\lambda p.\, p\,(\lambda i.\, \lambda j.\, i\, k))$
6. $\overline{\mathbf{p}_2\, M} = \lambda k.\, \overline{M}\,(\lambda p.\, p\,(\lambda i.\, \lambda j.\, j\, k))$
7. $\overline{\mathbf{k}_1\, M} = \lambda k.\, k\,(\lambda i.\, \lambda j.\, i\, \overline{M})$
8. $\overline{\mathbf{k}_2\, M} = \lambda k.\, k\,(\lambda i.\, \lambda j.\, j\, \overline{M})$
9. $\overline{\mathbf{D}_{x,y}(M, N, O)} = \lambda k.\, \overline{M}\,(\lambda m.\, m\,(\lambda x.\, \overline{N}\, k)\,(\lambda y.\, \overline{O}\, k))$

where k, m, p, i and j are fresh variables. ■

We now prove that the translations of Definition 4 and 5 commute with the typing relation.

Proposition 2. *Let M be a λ-term of $\lambda^{\rightarrow\wedge\vee}$ typable with type α under a set of declarations Γ. Then $\overline{M}$ is a λ-term of the simply typed λ-calculus, typable with type $\overline{\alpha}$ under the set of declarations $\overline{\Gamma}$.*

Proof. See Appendix A. □

The translation of Definition 5 does not map normal forms to normal forms. This is due to the so-called administrative redexes that are introduced by the translation. The modified translation below circumvents this problem.

Definition 6. *(Modified CPS-translation of $\lambda^{\rightarrow\wedge\vee}$) The modified CPS-translation $\overline{\overline{M}}$ of any λ-term M of $\lambda^{\rightarrow\wedge\vee}$ is defined as:*

$$\overline{\overline{M}} = \lambda k.\,(M : k)$$

where k is a fresh variable, and where the infix operator ":" obeys the following definition:

1. $x : K = x\,K$
2. $\lambda x.\,M : K = K\,(\lambda x.\,\overline{\overline{M}})$
3. $(M\,N) : K = M : \lambda m.\,m\,\overline{\overline{N}}\,K$
4. $\mathbf{p}(M, N) : K = K\,(\lambda p.\,p\,\overline{\overline{M}}\,\overline{\overline{N}})$
5. $\mathbf{p}_1\,M : K = M : \lambda p.\,p\,(\lambda i.\,\lambda j.\,i\,K)$
6. $\mathbf{p}_2\,M : K = M : \lambda p.\,p\,(\lambda i.\,\lambda j.\,j\,K)$
7. $\mathbf{k}_1\,M : K = K\,(\lambda i.\,\lambda j.\,i\,\overline{\overline{M}})$
8. $\mathbf{k}_2\,M : K = K\,(\lambda i.\,\lambda j.\,j\,\overline{\overline{M}})$
9. $\mathbf{D}_{x,y}(M, N, O) : K = M : \lambda m.\,m\,(\lambda x.\,(N : K))\,(\lambda y.\,(O : K))$

where m, p, i and j are fresh variables, and where, in Clause 9, x and y do not occur free in K. Remark that this last condition is not restrictive since it may always be satisfied by renaming. ■

As expected, the modified translation is a β-reduced form of the CPS-translation.

Lemma 3. *Let M and K be terms of $\lambda^{\rightarrow\wedge\vee}$. Then:*

1. $\overline{M} \twoheadrightarrow_\beta \overline{\overline{M}}$,
2. $\overline{M}\,K \twoheadrightarrow_\beta M : K$.

Proof. We proceed by induction on the structure of M. Property 1 is the property of interest, while Property 2 is needed to make the induction work. □

From this lemma, we get the analogue of Proposition 2 for the modified translation.

Proposition 3. *Let M be a λ-term of $\lambda^{\rightarrow\wedge\vee}$ typable with type α under a set of declarations Γ. Then $\overline{\overline{M}}$ is a λ-term of the simply typed λ-calculus, typable with type $\overline{\alpha}$ under the set of declarations $\overline{\Gamma}$.*

Proof. The proposition follows from Proposition 2, Lemma 3, and the subject reduction property of the simply typed λ-calculus. □

The modified CPS-translation allows the detour-conversions of $\lambda^{\to\wedge\vee}$ to be simulated by β-reduction. This is established by the next lemmas.

Lemma 4. *Let M and N be λ-terms of $\lambda^{\to\wedge\vee}$ and K be a simple λ-term in which there is no free occurrence of x. Then:*

1. $(M : K)[x{:=}\overline{\overline{N}}] \twoheadrightarrow_\beta (M[x{:=}N]) : K$,
2. $\overline{\overline{M}}[x{:=}\overline{\overline{N}}] \twoheadrightarrow_\beta \overline{\overline{M[x{:=}N]}}$.

Proof. *Property 2 is a direct consequence of Property 1, which is established by a straightforward induction on the structure of M.* □

Lemma 5. *Let K and L be simple λ-terms such that $K \xrightarrow{+}_\beta L$. Then, for any λ-term M of $\lambda^{\to\wedge\vee}$, $M : K \xrightarrow{+}_\beta M : L$.*

Proof. *By a straightforward induction on the structure of M.* □

Lemma 6. *Let M and N be two λ-terms of $\lambda^{\to\wedge\vee}$ such that $M \to_D N$. Then:*

1. $M : K \xrightarrow{+}_\beta N : K$, *for any simple λ-term K,*
2. $\overline{\overline{M}} \xrightarrow{+}_\beta \overline{\overline{N}}$.

Proof. *Property 2 may be established as a direct consequence of Property 1. Then proving that $C[M] : K \xrightarrow{+}_\beta C[N] : K$ (for any K) whenever $M : K \xrightarrow{+}_\beta N : K$ (for any K) consists in a straightforward induction on the structure of the context $C[\,]$, using Lemma 5 in some cases. Hence it remains to establish Property 1 for the five rewriting rules of Definition 1.*

$$
\begin{aligned}
((\lambda x.\,M)\,N) : K &= (\lambda x.\,M) : \lambda m.\,m\,\overline{\overline{N}}\,K \\
&= (\lambda m.\,m\,\overline{\overline{N}}\,K)\,(\lambda x.\,\overline{\overline{M}}) \\
&\to_\beta (\lambda x.\,\overline{\overline{M}})\,\overline{\overline{N}}\,K \\
&\to_\beta \overline{\overline{M}}[x{:=}\overline{\overline{N}}]\,K \\
&\twoheadrightarrow_\beta \overline{\overline{M[x{:=}N]}}\,K \\
&\twoheadrightarrow_\beta (M[x{:=}N]) : K
\end{aligned}
$$

$$
\begin{aligned}
(\mathbf{p}_i\,\mathbf{p}(M, N)) : K &= \mathbf{p}(M_1, M_2) : \lambda p.\,p\,(\lambda j_1.\,\lambda j_2.\,j_i\,K) \\
&= (\lambda p.\,p\,(\lambda j_1.\,\lambda j_2.\,j_i\,K))\,(\lambda p.\,p\,\overline{\overline{M}}_1\,\overline{\overline{M}}_2) \\
&\to_\beta (\lambda p.\,p\,\overline{\overline{M}}_1\,\overline{\overline{M}}_2)\,(\lambda j_1.\,\lambda j_2.\,j_i\,K) \\
&\to_\beta (\lambda j_1.\,\lambda j_2.\,j_i\,K)\,\overline{\overline{M}}_1\,\overline{\overline{M}}_2 \\
&\xrightarrow{+}_\beta \overline{\overline{M}}_i\,K \\
&\twoheadrightarrow_\beta M_i : K
\end{aligned}
$$

$$
\begin{aligned}
&\mathbf{D}_{x_1,x_2}(\mathbf{k}_i\, M, N_1, N_2) : K \\
&\qquad = (\mathbf{k}_i\, M) : \lambda m.\, m\, (\lambda x_1.\, (N_1 : K))\, (\lambda x_2.\, (N_2 : K)) \\
&\qquad = (\lambda m.\, m\, (\lambda x_1.\, (N_1 : K))\, (\lambda x_2.\, (N_2 : K)))\, (\lambda j_1.\, \lambda j_2.\, j_i\, \overline{\overline{M}}) \\
&\qquad \to_\beta (\lambda j_1.\, \lambda j_2.\, j_i\, \overline{\overline{M}})\, (\lambda x_1.\, (N_1 : K))\, (\lambda x_2.\, (N_2 : K)) \\
&\qquad \xrightarrow{+}_\beta (\lambda x_i.\, (N_i : K))\, \overline{\overline{M}} \\
&\qquad \to_\beta (N_i : K)[x_i{:=}\overline{\overline{M}}] \\
&\qquad \twoheadrightarrow_\beta (N_i[x_i{:=}M]) : K
\end{aligned}
$$

□

The above lemma allows any sequence of detour-conversion steps to be simulated by a longer sequence of β-reduction steps in the simply typed λ-calculus. Therefore, since the simply typed λ-calculus is strongly β-normalisable, we immediately obtain the following proposition.

Proposition 4. *$\lambda^{\to\wedge\vee}$ is strongly normalisable with respect to the detour-conversions.* □

5 Strong normalisation

In this section we prove that $\lambda^{\to\wedge\vee}$ is strongly normalisable with respect to both the detour- and permutation-conversions. This is not a direct consequence of Propositions 1 and 4 because detour-conversions can create permutation-redexes and, conversely, permutation conversions can create detour-redexes. Therefore we first show that the modified CPS-translation maps the relation of permutation-conversion to syntactic equality.

Lemma 7. *Let M and N be two λ-terms of $\lambda^{\to\wedge\vee}$ such that $M \to_P N$. Then:*

1. *$M : K = N : K$, for any simple λ-term K,*
2. *$\overline{\overline{M}} = \overline{\overline{N}}$.*

Proof. Property 2 is a direct consequence of Property 1. To show that $C[M] : K = C[N] : K$ whenever $M : K = N : K$ consists in a routine induction on the structure of the context $C[\,]$. It remains to establish Property 1 for the four rewriting rules of Definition 2.

$$
\begin{aligned}
&(\mathbf{D}_{x,y}(M, N, O)\, P) : K \\
&\qquad = \mathbf{D}_{x,y}(M, N, O) : \lambda m.\, m\, \overline{\overline{P}}\, K \\
&\qquad = M : \lambda m.\, m\, (\lambda x.\, (N : \lambda m.\, m\, \overline{\overline{P}}\, K))\, (\lambda y.\, (O : \lambda m.\, m\, \overline{\overline{P}}\, K)) \\
&\qquad = M : \lambda m.\, m\, (\lambda x.\, (N\, P : K))\, (\lambda y.\, (O\, P : K)) \\
&\qquad = \mathbf{D}_{x,y}(M, N\, P, O\, P) : K
\end{aligned}
$$

$$
\begin{aligned}
(\mathbf{p}_i\, \mathbf{D}_{x,y}(M, N, O)) : K &= \mathbf{D}_{x,y}(M, N, O) : \lambda p.\, p\, (\lambda j_1.\, \lambda j_2.\, j_i\, K) \\
&= M : \lambda m.\, m\, (\lambda x.\, (N : \lambda p.\, p\, (\lambda j_1.\, \lambda j_2.\, j_i\, K))) \\
&\qquad\qquad (\lambda y.\, (O : \lambda p.\, p\, (\lambda j_1.\, \lambda j_2.\, j_i\, K))) \\
&= M : \lambda m.\, m\, (\lambda x.\, (\mathbf{p}_i\, N : K))\, (\lambda y.\, (\mathbf{p}_i\, O : K)) \\
&= \mathbf{D}_{x,y}(M, \mathbf{p}_i\, N, \mathbf{p}_i\, O) : K
\end{aligned}
$$

$$
\begin{aligned}
&\mathbf{D}_{u,v}(\mathbf{D}_{x,y}(M,N,O),P,Q) : K \\
&\qquad = \mathbf{D}_{x,y}(M,N,O) : \lambda m.\, m\,(\lambda u.\,(P : K))\,(\lambda v.\,(Q : K)) \\
&\qquad = M : \lambda m.\, m\,(\lambda x.\,(N : \lambda m.\, m\,(\lambda u.\,(P : K))\,(\lambda v.\,(Q : K)))) \\
&\qquad\qquad (\lambda y.\,(O : \lambda m.\, m\,(\lambda u.\,(P : K))\,(\lambda v.\,(Q : K)))) \\
&\qquad = M : \lambda m.\, m\,(\lambda x.\,(\mathbf{D}_{u,v}(N,P,Q) : K))\,(\lambda y.\,(\mathbf{D}_{u,v}(O,P,Q) : K)) \\
&\qquad = \mathbf{D}_{x,y}(M, \mathbf{D}_{u,v}(N,P,Q), \mathbf{D}_{u,v}(O,P,Q)) : K
\end{aligned}
$$

□

We may now prove the main result.

Theorem 1. $\lambda^{\to\wedge\vee}$ *is strongly normalisable with respect to the reduction relation induced by the union of the detour- and permutation-conversions.*

Proof. Suppose it is not the case. Then there would exist an infinite sequence of detour- and permutation-conversion steps starting from a typable term (say, M) of $\lambda^{\to\wedge\vee}$. If this infinite sequence contains infinitely many detour-conversion steps, there must exist, by Lemmas 6 and 7 an infinite sequence of β-reduction steps starting from $\overline{\overline{\overline{M}}}$. But this, by Proposition 3, would contradict the strong normalisation of the simply typed λ-calculus. Hence the infinite sequence may contain only a finite number of detour conversion steps. But then, it would contain an infinite sequence of consecutive permutation-conversion steps, which contradicts Proposition 1. □

6 First-order quantification

As we stressed in the introduction, our proof of the strong normalisation of $\lambda^{\to\wedge\vee}$ (with the permuation-conversions) is modular. Consequently, it has the advantage of being easily adaptable. We illustrate this fact by taking into account intuitionistic positive first-order logic (**IPFOL**, for short), i.e., **IPPL** + universal and existential quantification. We first complete the grammar of terms with new constructors corresponding to the introduction and elimination of the two quantifiers:[2]

$$\mathcal{T} ::= \ \ldots \mid \mathbf{A_I}\,\mathcal{T} \mid \mathbf{A_E}\,\mathcal{T} \mid \mathbf{i}\,\mathcal{T} \mid \mathbf{E}_{\mathcal{X}}(\mathcal{T},\mathcal{T}) \mid$$

The typing rules of these new constructions correspond to the introduction and elimination rules of ∀ and ∃:

$$
\frac{M : \alpha[\xi]}{\mathbf{A_I}\,M : \forall\xi.\alpha[\xi]}
\qquad
\frac{M : \forall\xi.\alpha[\xi]}{\mathbf{A_E}\,M : \alpha[t]}
\qquad
\frac{M : \alpha[t]}{\mathbf{i}\,M : \exists\xi.\alpha[\xi]}
\qquad
\frac{M : \exists\xi.\alpha[\xi] \qquad \begin{array}{c}[\,x : \alpha[\xi]\,]\\ N : \beta\end{array}}{\mathbf{E}_x(M,N) : \beta}
$$

The resulting system ($\lambda^{\to\wedge\vee\forall\exists}$) features two additional detour-conversion rules.

[2] We have simplified the syntax with respect to [14]. The reason is that we are not interested in a complete coding of the proofs, but only in a coding that simulates proof normalisation. Consequently, there is no need for encoding the management of first-order terms.

Definition 7. *(Detour-conversions of $\lambda^{\to\wedge\vee\forall\exists}$) The detour-conversions of $\lambda^{\to\wedge\vee\forall\exists}$ consists of the five conversion rules of Definition 1, together with the two following rules:*

6. $\mathbf{A_E}\,\mathbf{A_I}\,M \to_D M$
7. $\mathbf{E}_x(\mathbf{i}\,M, N) \to_D N[x{:=}M]$ ∎

We must also add to the system two permutation-conversion rules obeying the general scheme of Section 2.4.

Definition 8. *(Permutation-conversions of $\lambda^{\to\wedge\vee\forall\exists}$) The permutation-conversions of $\lambda^{\to\wedge\vee\forall\exists}$ consists of the four conversion rules of Definition 2, together with the two following rules:*

5. $\mathbf{A_E}\,\mathbf{D}_{x,y}(M, N, O) \to_P \mathbf{D}_{x,y}(M, \mathbf{A_E}\,N, \mathbf{A_E}\,O)$
6. $\mathbf{E}_z(\mathbf{D}_{x,y}(M, N, O), P) \to_P \mathbf{D}_{x,y}(M, \mathbf{E}_z(N, P), \mathbf{E}_z(O, P))$ ∎

These four additional conversion rules are not sufficient to guarantee that the normal proofs satisfy the subformula property. Indeed the elimination rule for the existential quantifier presents the same peculiarity as the elimination rule for disjunction: its conclusion is not a subformula of the formula to be eliminated. Consequently, we must also allow all the elimination rules to permute with the existential elimination rule. This gives rise to the following *existential* permutation-conversions.

Definition 9. *(Existential permutation-conversions of $\lambda^{\to\wedge\vee\forall\exists}$)*

1. $\mathbf{E}_x(M, N)\,O \to_{P_\exists} \mathbf{E}_x(M, N\,O)$
2. $\mathbf{p}_1\,\mathbf{E}_x(M, N) \to_{P_\exists} \mathbf{E}_x(M, \mathbf{p}_1\,N)$
3. $\mathbf{p}_2\,\mathbf{E}_x(M, N) \to_{P_\exists} \mathbf{E}_x(M, \mathbf{p}_2\,N)$
4. $\mathbf{D}_{y,z}(\mathbf{E}_x(M, N), O, P) \to_{P_\exists} \mathbf{E}_x(M, \mathbf{D}_{y,z}(N, O, P))$
5. $\mathbf{A_E}\,\mathbf{E}_x(M, N) \to_{P_\exists} \mathbf{E}_x(M, \mathbf{A_E}\,N)$
6. $\mathbf{E}_y(\mathbf{E}_x(M, N), O) \to_{P_\exists} \mathbf{E}_x(M, \mathbf{E}_y(N, O))$ ∎

In order to prove that $\lambda^{\to\wedge\vee\forall\exists}$ satisfies the strong normalisation property with respect to the reduction relation induced by $\to_D$, $\to_P$ and$\to_{P_\exists}$, it suffices to adapt the different propositions of the previous sections. To this end, we generalise the notions of permutation degree, negative translation, CPS-translation, and modified CPS-translation.

Definition 10. *(Permutation degree for $\lambda^{\to\wedge\vee\forall\exists}$) The notion of permutation degree is adapted to $\lambda^{\to\wedge\vee\forall\exists}$ by adding the following clauses to Definition 3.*

19. $|\mathbf{A_I}\,M| = |M|$
20. $|\mathbf{A_E}\,M| = |M| + \#M$
21. $|\mathbf{i}\,M| = |M|$
22. $|\mathbf{E}_x(M, N)| = |M| + \#M \times |N|$
23. $\#\mathbf{A_I}\,M = 1$
24. $\#\mathbf{A_E}\,M = \#M$

25. $\#\mathbf{i}\,M = 1$
26. $\#\mathbf{E}_x(M,N) = 2 \times \#M \times \#N$ ∎

Lemma 8. *Let M and N be two λ-terms of $\lambda^{\to\wedge\vee\forall\exists}$ such that $M \to_P N$. Then $\#M = \#N$.*

Proof. *We extend the proof of Lemma 1 by showing that # is invariant under the additional rewriting rule of Definition 8.*

$$\begin{aligned}\#\mathbf{A_E}\,\mathbf{D}_{x,y}(M,N,O) &= \#\mathbf{D}_{x,y}(M,N,O)\\ &= 2 \times \#M \times (\#N + \#O)\\ &= 2 \times \#M \times (\#\mathbf{A_E}\,N + \#\mathbf{A_E}\,O)\\ &= \#\mathbf{D}_{x,y}(M, \mathbf{A_E}\,N, \mathbf{A_E}\,O)\end{aligned}$$

$$\begin{aligned}\#\mathbf{E}_z(\mathbf{D}_{x,y}(M,N,O),P) &= 2 \times \#\mathbf{D}_{x,y}(M,N,O) \times \#P\\ &= 4 \times \#M \times (\#N + \#O) \times \#P\\ &= 2 \times \#M \times (2 \times \#N \times \#P + 2 \times \#O \times \#P)\\ &= 2 \times \#M \times (\#\mathbf{E}_z(N,P) + \#\mathbf{E}_z(O,P))\\ &= \#\mathbf{D}_{x,y}(M, \mathbf{E}_z(N,P), \mathbf{E}_z(O,P))\end{aligned}$$

□

Lemma 9. *Let M and N be two λ-terms of $\lambda^{\to\wedge\vee\forall\exists}$ such that $M \to_P N$. Then $|M| > |N|$.*

Proof. *We extend the proof of Lemma 2 as follows:*

$$\begin{aligned}|\mathbf{A_E}\,\mathbf{D}_{x,y}(M,N,O)| &= |\mathbf{D}_{x,y}(M,N,O)| + \#\mathbf{D}_{x,y}(M,N,O)\\ &= |M| + \#M \times (|N| + |O|) + 2 \times \#M \times (\#N + \#O)\\ &> |M| + \#M \times (|N| + |O|) + \#M \times (\#N + \#O)\\ &= |M| + \#M \times (|N| + \#N + |M| + \#M)\\ &= |\mathbf{D}_{x,y}(M, \mathbf{A_E}\,N, \mathbf{A_E}\,O)|\end{aligned}$$

$$\begin{aligned}&|\mathbf{E}_z(\mathbf{D}_{x,y}(M,N,O),P)|\\ &\quad= |\mathbf{D}_{x,y}(M,N,O)| + \#\mathbf{D}_{x,y}(M,N,O) \times \#P\\ &\quad= |M| + \#M \times (|N| + |O|) + 2 \times \#M \times (\#N + \#O)) \times \#P\\ &\quad> |M| + \#M \times (|N| + |O|) + \#M \times (\#N + \#O)) \times \#P\\ &\quad= |M| + \#M \times (|N| + \#N \times \#P + |O| + \#O \times \#P)\\ &\quad= |M| + \#M \times (|\mathbf{E}_z(N,P)| + |\mathbf{E}_z(O,P)|)\\ &\quad= |\mathbf{D}_{x,y}(M, \mathbf{E}_z(N,P), \mathbf{E}_z(O,P))|\end{aligned}$$

□

Lemma 10. *Let M and N be two λ-terms of $\lambda^{\to\wedge\vee\forall\exists}$ such that $M \to_{P_\exists} N$. Then $\#M = \#N$.*

Proof. *The proof is similar to that of Lemma 1. We show that # is invariant under the six rules of Definition 9:*

$$\begin{aligned}\#\mathbf{E}_x(M,N)\,O &= \#\mathbf{E}_x(M,N)\\ &= 2 \times \#M \times \#N\\ &= 2 \times \#M \times \#N\,O\\ &= \#\mathbf{E}_x(M, N\,O)\end{aligned}$$

$$
\begin{aligned}
\#\mathbf{p}_i\,\mathbf{E}_x(M,N) &= \#\mathbf{E}_x(M,N)\\
&= 2\times\#M\times\#N\\
&= 2\times\#M\times\#\mathbf{p}_i\,N\\
&= \#\mathbf{E}_x(M,\mathbf{p}_i\,N)
\end{aligned}
$$

$$
\begin{aligned}
\#\mathbf{D}_{y,z}(\mathbf{E}_x(M,N),O,P) &= 2\times\#\mathbf{E}_x(M,N)\times(\#O+\#P)\\
&= 4\times\#M\times\#N\times(\#O+\#P)\\
&= 2\times\#M\times\#\mathbf{D}_{y,z}(N,O,P)\\
&= \#\mathbf{E}_x(M,\mathbf{D}_{y,z}(N,O,P))
\end{aligned}
$$

$$
\begin{aligned}
\#\mathbf{A}_\mathbf{E}\,\mathbf{E}_x(M,N) &= \#\mathbf{E}_x(M,N)\\
&= 2\times\#M\times\#N\\
&= 2\times\#M\times\#\mathbf{A}_\mathbf{E}\,N\\
&= \#\mathbf{E}_x(M,\mathbf{A}_\mathbf{E}\,N)
\end{aligned}
$$

$$
\begin{aligned}
\#\mathbf{E}_y(\mathbf{E}_x(M,N),O) &= 2\times\#\mathbf{E}_x(M,N)\times\#O\\
&= 4\times\#M\times\#N\times\#O\\
&= 2\times\#M\times\#\mathbf{E}_y(N,O)\#\mathbf{E}_x(M,\mathbf{E}_y(N,O))
\end{aligned}
$$

□

Lemma 11. *Let M and N be two λ-terms of $\lambda^{\rightarrow\wedge\vee\forall\exists}$ such that $M \rightarrow_{P_\exists} N$. Then $|M| > |N|$.*

Proof. *The proof is similar to that of Lemma 2:*

$$
\begin{aligned}
|\mathbf{E}_x(M,N)\,O| &= |\mathbf{E}_x(M,N)| + \#\mathbf{E}_x(M,N)\times|O|\\
&= |M| + \#M\times|N| + 2\times\#M\times\#N\times|O|\\
&> |M| + \#M\times|N| + \#M\times\#N\times|O|\\
&= |M| + \#M\times(|N| + \#N\times|O|)\\
&= |M| + \#M\times|N\,O|\\
&= |\mathbf{E}_x(M,N\,O)|
\end{aligned}
$$

$$
\begin{aligned}
|\mathbf{p}_i\,\mathbf{E}_x(M,N)| &= |\mathbf{E}_x(M,N)| + \#\mathbf{E}_x(M,N)\\
&= |M| + \#M\times|N| + 2\times\#M\times\#N\\
&> |M| + \#M\times|N| + \#M\times\#N\\
&= |M| + \#M\times(|N| + \#N)\\
&= |M| + \#M\times|\mathbf{p}_i\,N|\\
&= |\mathbf{E}_x(M,\mathbf{p}_i\,N)|
\end{aligned}
$$

$$
\begin{aligned}
|\mathbf{D}_{y,z}(\mathbf{E}_x(M,N),O,P)| &= |\mathbf{E}_x(M,N)| + \#\mathbf{E}_x(M,N)\times(|O|+|P|)\\
&= |M| + \#M\times|N| + 2\times\#M\times\#N\times(|O|+|P|)\\
&> |M| + \#M\times|N| + \#M\times\#N\times(|O|+|P|)\\
&= |M| + \#M\times(|N| + \#N\times(|O|+|P|))\\
&= |M| + \#M\times|\mathbf{D}_{y,z}(N,O,P)|\\
&= |\mathbf{E}_x(M,\mathbf{D}_{y,z}(N,O,P))|
\end{aligned}
$$

$$
\begin{aligned}
|\mathbf{A_E}\,\mathbf{E}_x(M,N)| &= |\mathbf{E}_x(M,N)| + \#\mathbf{E}_x(M,N) \\
&= |M| + \#M \times |N| + 2 \times \#M \times \#N \\
&> |M| + \#M \times |N| + \#M \times \#N \\
&= |M| + \#M \times (|N| + \#N) \\
&= |M| + \#M \times |\mathbf{A_E}\,N| \\
&= |\mathbf{E}_x(M, \mathbf{A_E}\,N)|
\end{aligned}
$$

$$
\begin{aligned}
|\mathbf{E}_y(\mathbf{E}_x(M,N),O)| &= |\mathbf{E}_x(M,N)| + \#\mathbf{E}_x(M,N) \times |O| \\
&= |M| + \#M \times |N| + 2 \times \#M \times \#N \times |O| \\
&> |M| + \#M \times |N| + \#M \times \#N \times |O| \\
&= |M| + \#M \times (|N| + \#N \times |O|) \\
&= |M| + \#M \times |\mathbf{E}_y(N,O)| \\
&= |\mathbf{E}_x(M, \mathbf{E}_y(N,O))|
\end{aligned}
$$

□

As an immediate consequence of Lemmas 9 and 11, We obtain the following proposition.

Proposition 5. $\lambda^{\to\wedge\vee\forall\exists}$ *is strongly normalisable with respect to the reduction relation induced by the permutation-, and existential permutation-conversions.*

□

The next step in establishing the strong normalisation of $\lambda^{\to\wedge\vee\forall\exists}$ with respect to all the rduction relations is to generalise the translations of Definitions 4 and 5. To this end, we let ι be a distinguished atomic simple type, and $\star$ be a distinguished variable of type ι.

Definition 11. *(Negative translation of* **IPFOL***) The definition of the negative translation $\overline{\alpha}$ of any formula α of* **IPFOL** *is obtained by adding the following equations to Definition 4:*

5. $a(\xi_1, \ldots, \xi_n)^\circ = a$
6. $(\forall\xi.\alpha)^\circ = \iota \to \overline{\alpha}$
7. $(\exists\xi.\alpha)^\circ = {\sim}(\iota \to ({\sim}\overline{\alpha}))$

Definition 12. *(CPS-translation of $\lambda^{\to\wedge\vee\forall\exists}$) The CPS-translation of Definition 5 is extended to $\lambda^{\to\wedge\vee\forall\exists}$ by adding the following equations:*

10. $\overline{\mathbf{A_I}\,M} = \lambda k.\,k\,(\lambda\xi.\,\overline{M})$, *where ξ is a fresh variable.*
11. $\overline{\mathbf{A_E}\,M} = \lambda k.\,\overline{M}\,(\lambda m.\,m \star k)$
12. $\overline{\mathbf{i}\,M} = \lambda k.\,k\,(\lambda a.\,a \star \overline{M})$
13. $\overline{\mathbf{E}_x(M,N)} = \lambda k.\,\overline{M}\,(\lambda m.\,m\,(\lambda\xi.\,\lambda x.\,\overline{N}\,k))$, *where ξ is a fresh variable.* ■

The commutation property established by Proposition 2 still holds with the extended definitions.

Proposition 6. *Let M be a λ-term of $\lambda^{\to\wedge\vee\forall\exists}$ typable with type α under a set of declarations Γ. Then $\overline{M}$ is a λ-term of the simply typed λ-calculus, typable with type $\overline{\alpha}$ under the set of declarations $\overline{\Gamma}$.*

Proof. See Appendix A. □

The definition of the modified CPS-translation is generalised as follows.

Definition 13. *(Modified CPS-translation of $\lambda^{\to\wedge\vee\forall\exists}$) The modified CPS-translation $\overline{\overline{M}}$ of any λ-term M of $\lambda^{\to\wedge\vee\forall\exists}$ is defined by adding the following equations to Definition 6:*

10. $\mathbf{A_I}\, M : K = K\,(\lambda\xi.\,\overline{\overline{M}})$, *where ξ is a fresh variable.*
11. $\mathbf{A_E}\, M : K = M : \lambda m.\, m \star K$
12. $\mathbf{i}\, M : K = K\,(\lambda a.\, a \star \overline{\overline{M}})$
13. $\mathbf{E}_x(M, N) : K = M : \lambda m.\, m\,(\lambda\xi.\,\lambda x.\,(N : K))$, *where ξ is a fresh variable.*

■

Showing that $\overline{M} \twoheadrightarrow_\beta \overline{\overline{M}}$ for any λ-term M of $\lambda^{\to\wedge\vee\forall\exists}$ is a matter of routine. From this, we get that the negative translation of Definition 11 and the modified CPS-translation of Definition 13 commute with the typing relation.

Proposition 7. *Let M be a λ-term of $\lambda^{\to\wedge\vee\forall\exists}$ typable with type α under a set of declarations Γ. Then $\overline{\overline{M}}$ is a λ-term of the simply typed λ-calculus, typable with type $\overline{\alpha}$ under the set of declarations $\overline{\Gamma}$.* □

It remains to prove that that the modified CPS-translation allows the detour-conversion to be simulated by β-reduction, and the other reduction relations to be simulated by equality. We state the main lemmas for the sake of completeness.

Lemma 12. *Let M and N be two λ-terms of $\lambda^{\to\wedge\vee\forall\exists}$ such that $M \to_D N$. Then:*

1. $M : K \xrightarrow{+}_\beta N : K$, *for any simple λ-term K,*
2. $\overline{\overline{M}} \xrightarrow{+}_\beta \overline{\overline{N}}$.

Proof. We extend the proof of Lemma 6 as follows:

$$\begin{aligned}
(\mathbf{A_E}\,\mathbf{A_I}\,M) : K &= (\mathbf{A_I}\,M) : \lambda m.\, m \star K \\
&= (\lambda m.\, m \star K)\,(\lambda\xi.\,\overline{\overline{M}}) \\
&\to_\beta (\lambda\xi.\,\overline{\overline{M}}) \star K \\
&\to_\beta \overline{\overline{M}}\,K \\
&\twoheadrightarrow_\beta M : K \\
\mathbf{E}_x(\mathbf{i}\,M, N) : K &= (\mathbf{i}\,M) : \lambda m.\, m\,(\lambda\xi.\,\lambda x.\,(N : K)) \\
&= (\lambda m.\, m\,(\lambda\xi.\,\lambda x.\,(N : K)))\,(\lambda a.\, a \star \overline{\overline{M}}) \\
&\to_\beta (\lambda a.\, a \star \overline{\overline{M}})\,(\lambda\xi.\,\lambda x.\,(N : K)) \\
&\to_\beta (\lambda\xi.\,\lambda x.\,(N : K)) \star \overline{\overline{M}} \\
&\twoheadrightarrow_\beta (\lambda x.\,(N : K))\,\overline{\overline{M}} \\
&\twoheadrightarrow_\beta (N : K)[x{:=}\overline{\overline{M}}] \\
&\twoheadrightarrow_\beta N[x{:=}M] : K
\end{aligned}$$

□

Lemma 13. *Let M and N be two λ-terms of $\lambda^{\to\wedge\vee\forall\exists}$ such that $M \to_P N$. Then:*

1. *$M : K = N : K$, for any simple λ-term K,*
2. *$\overline{\overline{M}} = \overline{\overline{N}}$.*

Proof. We extend the proof of Lemma 7.

$$\begin{aligned}(\mathbf{A_E}\,\mathbf{D}_{x,y}(M,N,O)) : K &= \mathbf{D}_{x,y}(M,N,O) : \lambda m.\, m \star K\\ &= M : \lambda m.\, m\,(\lambda x.\,(N : \lambda m.\, m \star K))\\ &\qquad\qquad (\lambda y.\,(O : \lambda m.\, m \star K))\\ &= M : \lambda m.\, m\,(\lambda x.\,((\mathbf{A_E}\,N) : K))\,(\lambda y.\,((\mathbf{A_E}\,O) : K))\\ &= \mathbf{D}_{x,y}(M, \mathbf{A_E}\,N, \mathbf{A_E}\,O) : K\end{aligned}$$

$$\begin{aligned}\mathbf{E}_z(\mathbf{D}_{x,y}(M,N,O),P) : K &= \mathbf{D}_{x,y}(M,N,O) : \lambda m.\, m\,(\lambda \xi.\,\lambda x.\,(P : K))\\ &= M : \lambda m.\, m\,(\lambda x.\,(N : \lambda m.\, m\,(\lambda \xi.\,\lambda x.\,(P : K))))\\ &\qquad\qquad (\lambda y.\,(O : \lambda m.\, m\,(\lambda \xi.\,\lambda x.\,(P : K))))\\ &= M : \lambda m.\, m\,(\lambda x.\,(\mathbf{E}_z(N,P) : K))\\ &\qquad\qquad (\lambda y.\,(\mathbf{E}_z(O,P) : K))\\ &= \mathbf{D}_{x,y}(M, \mathbf{E}_z(N,P), \mathbf{E}_z(O,P)) : K\end{aligned}$$

□

Lemma 14. *Let M and N be two λ-terms of $\lambda^{\to\wedge\vee\forall\exists}$ such that $M \to_{P_\exists} N$. Then:*

1. *$M : K = N : K$, for any simple λ-term K,*
2. *$\overline{\overline{M}} = \overline{\overline{N}}$.*

Proof. We show that Property 1 holds for the six rewriting rules of Definition 9.

$$\begin{aligned}\mathbf{E}_x(M,N)\,O) : K &= \mathbf{E}_x(M,N) : \lambda m.\, m\,\overline{\overline{O}}\,K\\ &= M : \lambda m.\, m\,(\lambda \xi.\,\lambda x.\,(N : \lambda m.\, m\,\overline{\overline{O}}\,K))\\ &= M : \lambda m.\, m\,(\lambda \xi.\,\lambda x.\,((N\,O) : K))\\ &= \mathbf{E}_x(M, N\,O) : K\end{aligned}$$

$$\begin{aligned}(\mathbf{p}_i\,\mathbf{E}_x(M,N)) : K &= \mathbf{E}_x(M,N) : \lambda p.\, p\,(\lambda j_1.\,\lambda j_2.\, j_i\,K)\\ &= M : \lambda m.\, m\,(\lambda \xi.\,\lambda x.\,(N : \lambda p.\, p\,(\lambda j_1.\,\lambda j_2.\, j_i\,K)))\\ &= M : \lambda m.\, m\,(\lambda \xi.\,\lambda x.\,((\mathbf{p}_i\,N) : K))\\ &= \mathbf{E}_x(M, \mathbf{p}_i\,N) : K\end{aligned}$$

$$\begin{aligned}\mathbf{D}_{y,z}(\mathbf{E}_x(M,N),O,P) : K &= \mathbf{E}_x(M,N) : \lambda m.\, m\,(\lambda y.\,(O : K))\,(\lambda z.\,(P : K))\\ &= M : \lambda m.\, m\,(\lambda \xi.\,\lambda x.\,(N : \lambda m.\, m\,(\lambda y.\,(O : K))\\ &\qquad\qquad (\lambda z.\,(P : K))))\\ &= M : \lambda m.\, m\,(\lambda \xi.\,\lambda x.\,(\mathbf{D}_{y,z}(N,O,P) : K))\\ &= \mathbf{E}_x(M, \mathbf{D}_{y,z}(N,O,P)) : K\end{aligned}$$

$$\begin{aligned}(\mathbf{A_E}\,\mathbf{E}_x(M,N)) : K &= \mathbf{E}_x(M,N) : \lambda m.\, m \star K\\ &= M : \lambda m.\, m\,(\lambda \xi.\,\lambda x.\,(N : \lambda m.\, m \star K))\\ &= M : \lambda m.\, m\,(\lambda \xi.\,\lambda x.\,((\mathbf{A_E}\,N) : K))\\ &= \mathbf{E}_x(M, \mathbf{A_E}\,N) : K\end{aligned}$$

$$
\begin{aligned}
\mathbf{E}_y(\mathbf{E}_x(M,N),O) : K &= \mathbf{E}_x(M,N) : \lambda m.\, m\,(\lambda\xi.\, \lambda y.\, (O : K)) \\
&= M : \lambda m.\, m\,(\lambda\xi.\, \lambda x.\, (N : \lambda m.\, m\,(\lambda\xi.\, \lambda y.\, (O : K)))) \\
&= M : \lambda m.\, m\,(\lambda\xi.\, \lambda x.\, (\mathbf{E}_y(N,O) : K)) \\
&= \mathbf{E}_x(M, \mathbf{E}_y(N,O)) : K
\end{aligned}
$$

□

Propositions 5, 7, and Lemmas 12, 13, and 14 allow one to replay the proof of Theorem 1. Hence, we obtain the following theorem.

Theorem 2. $\lambda^{\rightarrow\wedge\vee\forall\exists}$ *is strongly normalisable with respect to the reduction relation induced by the union of the detour-, permutation-, and existential permutation-conversions.* □

7 Absurdity and ⊥-conversions

We now show how to extend our proof to full intuitionistic first-order logic (**IFOL**, for short), i.e., **IPFOL** + negation. Following Prawitz, we consider the negation as a defined connective:

$$\neg\alpha = \alpha \rightarrow \bot$$

where $\bot$ is a constant standing for absurdity that obeys the following elimination rule:

$$\frac{\bot}{\alpha}$$

Consequently, we extend the grammar of terms by the following clause:

$$\mathcal{T} ::= \ \ldots \mid \bot(\mathcal{T})$$

and we add the following typing rule to the system:

$$\frac{M : \bot}{\bot(M) : \alpha}$$

We call the resulting system $\lambda^{\rightarrow\wedge\vee\forall\exists\bot}$. Since there is no introduction rule for $\bot$, there is no detour-conversion either. On the other hand, we must take into account new permutation-conversion rules.

Definition 14. *(Permutation-conversions of* $\lambda^{\rightarrow\wedge\vee\forall\exists\bot}$*) The permutation-conversions of* $\lambda^{\rightarrow\wedge\vee\forall\exists\bot}$ *consists of the six conversion rules of Definition 8, together with the following rule:*

7. $\bot(\mathbf{D}_{x,y}(M,N,O)) \rightarrow_P \mathbf{D}_{x,y}(M,\bot(N),\bot(O))$ ■

Definition 15. *(Existential permutation-conversions of* $\lambda^{\rightarrow\wedge\vee\forall\exists\bot}$*) The existential permutation-conversions of* $\lambda^{\rightarrow\wedge\vee\forall\exists\bot}$ *consists of the six conversion rules of Definition 9, together with the following rule:*

7. $\bot(\mathbf{E}_x(M,N)) \to_{P_\exists} \mathbf{E}_x(M,\bot(N))$ ∎

To adapt our strong normalisation proof to $\lambda^{\to\wedge\vee\forall\exists\bot}$ we first complete the definition of permutation degree, and show that it is decreasing under the reduction relations of Definition 14 and 15.

Definition 16. *(Permutation degree for $\lambda^{\to\wedge\vee\forall\exists\bot}$) The notion of permutation degree is adapted to $\lambda^{\to\wedge\vee\forall\exists\bot}$ by adding the following clauses to Definition 10.*

27. $|\bot(M)| = |M| + \#M$
28. $\#\bot(M) = \#M$ ∎

Lemma 15. *Let M and N be two λ-terms of $\lambda^{\to\wedge\vee\forall\exists\bot}$ such that $M \to_P N$. Then $\#M = \#N$.*

Proof. *We extend the proof of Lemma 8 by showing that # is invariant under the additional rewriting rule of Definition 14.*

$$\begin{aligned}\#\bot(\mathbf{D}_{x,y}(M,N,O)) &= \#\mathbf{D}_{x,y}(M,N,O)\\ &= 2\times\#M\times(\#N+\#O)\\ &= 2\times\#M\times(\#\bot(N)+\#\bot(O))\\ &= \#\mathbf{D}_{x,y}(M,\bot(N),\bot(P))\end{aligned}$$

□

Lemma 16. *Let M and N be two λ-terms of $\lambda^{\to\wedge\vee\forall\exists\bot}$ such that $M \to_P N$. Then $|M| > |N|$.*

Proof. *We extend the proof of Lemma 9 as follows:*

$$\begin{aligned}|\bot(\mathbf{D}_{x,y}(M,N,O))| &= |\mathbf{D}_{x,y}(M,N,O)| + \#\mathbf{D}_{x,y}(M,N,O)\\ &= |M| + \#M\times(|N|+|O|) + 2\times\#M\times(\#N+\#O)\\ &> |M| + \#M\times(|N|+|O|) + \#M\times(\#N+\#O)\\ &= |M| + \#M\times(|N|+\#N+|O|+\#O)\\ &= |M| + \#M\times(|\bot(N)|+|\bot(O)|)\\ &= |\mathbf{D}_{x,y}(M,\bot(N),\bot(O))|\end{aligned}$$

□

Lemma 17. *Let M and N be two λ-terms of $\lambda^{\to\wedge\vee\forall\exists\bot}$ such that $M \to_{P_\exists} N$. Then $\#M = \#N$.*

Proof. *We extend the proof of Lemma 10 as follows:*

$$\begin{aligned}\#\bot(\mathbf{E}_x(M,N)) &= \#\mathbf{E}_x(M,N)\\ &= 2\times\#M\times\#N\\ &= 2\times\#M\times\#\bot(N)\\ &= \#\mathbf{E}_x(M,\bot(N))\end{aligned}$$

□

Lemma 18. *Let M and N be two λ-terms of $\lambda^{\to\wedge\vee\forall\exists\bot}$ such that $M \to_{P_\exists} N$. Then $|M| > |N|$.*

Proof. *We extend the proof of Lemma 11 as follows:*

$$
\begin{aligned}
|\bot(\mathbf{E}_x(M,N))| &= |\mathbf{E}_x(M,N)| + \#\mathbf{E}_x(M,N) \\
&= |M| + \#M \times |N| + 2 \times \#M \times \#N \\
&> |M| + \#M \times |N| + \#M \times \#N \\
&= |M| + \#M \times (|N| + \#N) \\
&= |M| + \#M \times |\bot N| \\
&= |\mathbf{E}_x(M, \bot(N))|
\end{aligned}
$$

□

Lemmas 16 and 18 immediately yield the following proposition.

Proposition 8. *$\lambda^{\to\wedge\vee\forall\exists\bot}$ is strongly normalisable with respect to the reduction relation induced by both the permutation- and existential permutation-conversions.* □

We now extend the negative translation of Definition 11 to **IFOL**, and the CPS-translation of Definition 12 to $\lambda^{\to\wedge\vee\forall\exists\bot}$.

Definition 17. *(Negative translation of* **IFOL***) The definition of the negative translation $\overline{\alpha}$ of any formula α of* **IFOL** *is obtained by adding the following equation to Definition 11:*

8. $\bot^\circ = o$ ■

Definition 18. *(CPS-translation of $\lambda^{\to\wedge\vee\forall\exists\bot}$) The CPS-translation of Definition 12 is extended to $\lambda^{\to\wedge\vee\forall\exists\bot}$ by adding the following equation:*

14. $\overline{\bot(M)} = \lambda k.\, \overline{M}\, ((\lambda x.\, \lambda y.\, y)\, k)$ ■

It might be surprising that the above clause introduces explicitly a β-redex, namely, $(\lambda x.\, \lambda y.\, y)\, k$. We will come back to this point later on.

Proposition 9. *Let M be a λ-term of $\lambda^{\to\wedge\vee\forall\exists\bot}$ typable with type α under a set of declarations Γ. Then $\overline{M}$ is a λ-term of the simply typed λ-calculus, typable with type $\overline{\alpha}$ under the set of declarations $\overline{\Gamma}$.*

Proof. *See Appendix A.* □

The definition of modified CPS-translation is then adapted as follows.

Definition 19. *(Modified CPS-translation of $\lambda^{\to\wedge\vee\forall\exists\bot}$) The modified CPS-translation $\overline{\overline{M}}$ of any λ-term M of $\lambda^{\to\wedge\vee\forall\exists\bot}$ is defined by adding the following equation to Definition 13:*

14. $\bot(M) : K = M : ((\lambda x.\, \lambda y.\, y)\, K)$ ■

Definitions 18 and 19 are such that $\overline{M} \twoheadrightarrow_\beta \overline{\overline{M}}$, from which we derive the following proposition.

Proposition 10. *Let M be a λ-term of $\lambda^{\to\wedge\vee\forall\exists\bot}$ typable with type α under a set of declarations Γ. Then $\overline{\overline{M}}$ is a λ-term of the simply typed λ-calculus, typable with type $\overline{\alpha}$ under the set of declarations $\overline{\Gamma}$.* □

It remains to prove that that the modified CPS-translation maps any two terms that are P- or $P_\exists$-convertible to the same simple λ-term.

Lemma 19. *Let M and N be two λ-terms of $\lambda^{\to\wedge\vee\forall\exists\bot}$ such that $M \to_P N$. Then:*

1. $M : K = N : K$, *for any simple λ-term K,*
2. $\overline{\overline{M}} = \overline{\overline{N}}$.

Proof. *The proof is adapted from that of Lemma 13. The additional case is handled as follows:*

$$\begin{aligned}
&\bot(\mathbf{D}_{x,y}(M,N,O)) : K\\
&\qquad = \mathbf{D}_{x,y}(M,N,O) : (\lambda x.\, \lambda y.\, y)\, K\\
&\qquad = M : \lambda m.\, m\, (\lambda x.\, (N : (\lambda x.\, \lambda y.\, y)\, K))\, (\lambda y.\, (O : (\lambda x.\, \lambda y.\, y)\, K))\\
&\qquad = M : \lambda m.\, m\, (\lambda x.\, (\bot(N) : K))\, (\lambda y.\, (\bot(O) : K))\\
&\qquad = (\mathbf{D}_{x,y}(M, \bot(N), \bot(O))) : K
\end{aligned}$$

□

Lemma 20. *Let M and N be two λ-terms of $\lambda^{\to\wedge\vee\forall\exists\bot}$ such that $M \to_{P_\exists} N$. Then:*

1. $M : K = N : K$, *for any simple λ-term K,*
2. $\overline{\overline{M}} = \overline{\overline{N}}$.

Proof. *The proof is adapted from that of Lemma 14. The additional case is handled as follows:*

$$\begin{aligned}
\bot(\mathbf{E}_x(M,N)) : K &= \mathbf{E}_x(M,N) : (\lambda x.\, \lambda y.\, y)\, K\\
&= M : \lambda m.\, m\, (\lambda \xi.\, \lambda x.\, (N : (\lambda x.\, \lambda y.\, y)\, K))\\
&= M : \lambda m.\, m\, (\lambda \xi.\, \lambda x.\, (\bot(N) : K))\\
&= \mathbf{E}_x(M, \bot(N)) : K
\end{aligned}$$

□

In order to conclude, it remains to prove that Lemma 12 is still valid. Since there is no new detour-conversion rule, we only have to check that $C[M] : K \xrightarrow{+}_\beta C[N] : K$ (for any K) whenever $M : K \xrightarrow{+}_\beta N : K$ (for any K). In order to prove this, we need Lemma 5, which indeed still holds because the infix operator ":" is strict in K. It is to be noted that the strictness of ":" is due to the redex $(\lambda x.\, \lambda y.\, y)\, K$ introduced by Definition 19. This explains why both Definitions 18 and 19 introduce explicitly β-redices. Finally, Lemma 12 being still valid, the following theorem holds.

Theorem 3. *$\lambda^{\to\wedge\vee\forall\exists\bot}$ is strongly normalisable with respect to the reduction relation induced by the union of the detour-, permutation- and existential permutation-conversions.* □

The above theorem is sufficient to ensure that the normal proofs satisfy the subformula property provided that all the formulas introduced by $\perp$-elimination are atomic. This may be assumed without loss of generality because any non-atomic instance of the $\perp$-elimination rule may be replaced by atomic ones. For example, in the case of a conjunctive formula, one may apply the following transformation:

$$\dfrac{\begin{array}{c}\vdots\ \Pi\\ \perp\end{array}}{\alpha\wedge\beta} \quad\longrightarrow\quad \dfrac{\dfrac{\begin{array}{c}\vdots\ \Pi\\ \perp\end{array}}{\alpha} \qquad \dfrac{\begin{array}{c}\vdots\ \Pi\\ \perp\end{array}}{\beta}}{\alpha\wedge\beta}$$

The use of such transformations is not completely satisfactory. On the one hand, it corresponds to a preprocessing of the proofs, which goes against the spirit of strong normalization. On the other hand, these transformations, when expressed in an untyped setting, do not normalize. For instance, the above transformation gives rise to the following non-terminating rewriting rule:

$$\perp(M) \to \mathbf{p}(\perp(M), \perp(M)).$$

These problems may be circumvented by taking into account conversion rules that are proper to $\perp$. These rules allow proofs to be simplified when the principal premise of an elimination rule is obtained as the conclusion of a $\perp$-elimination. For instance:

$$\dfrac{\dfrac{\begin{array}{c}\vdots\ \Pi_1\\ \perp\end{array}}{\alpha\to\beta} \qquad \begin{array}{c}\vdots\ \Pi_2\\ \alpha\end{array}}{\beta}\ \text{(Elim.)} \quad\longrightarrow\quad \dfrac{\begin{array}{c}\vdots\ \Pi_1\\ \perp\end{array}}{\beta}$$

From a computational point of view, these rules correspond to the propagation of a raised exception [4]. They are the following.

Definition 20. *($\perp$-conversions of $\lambda^{\to\wedge\vee\forall\exists\perp}$)*

1. $\perp(M)\,N \to_{\perp} \perp(M)$
2. $\mathbf{p}_1 \perp(M) \to_{\perp} \perp(M)$
3. $\mathbf{p}_2 \perp(M) \to_{\perp} \perp(M)$
4. $\mathbf{D}_{x,y}(\perp(M), N, O) \to_{\perp} \perp(M)$
5. $\mathbf{A_E} \perp(M) \to_{\perp} \perp(M)$
6. $\mathbf{E}_x(\perp(M), N) \to_{\perp} \perp(M)$
7. $\perp(\perp(M)) \to_{\perp} \perp(M)$ ■

It is possible to show that the permutation degree of Definition 16 is strictly decreasing under the above reduction rules. Unfortunately, it is not the case that the modified CPS-translation of Definition 19 maps two twerms that are

$\bot$-convertible to the same simply typed λ-term.[3] Hence, it is not possible to take the $\bot$-conversions into account by just replaying the proof of Theorem 1 (as we did for Theorems 2 and 3). Consequently, in order to show $\lambda^{\rightarrow\wedge\vee\forall\exists\bot}$ is strongly normalisable with respect to the reduction relation induced by the union of the detour-, permutation-, existential permutation-, and $\bot$-conversions, we adopt another simple strategy: we show that the $\bot$-conversions may be postponed.

First, note that the lengths of left-hand sides of the rewriting rules of Definition 20 are strictly less that the lengths of the corresponding right-hand sides. This simple observation immediately yields the following result.

Proposition 11. *$\lambda^{\rightarrow\wedge\vee\forall\exists\bot}$ is strongly normalisable with respect to the reduction relation induced by the $\bot$-conversions.* □

In order to establish the postponement of the $\bot$-conversions, the two following lemmas are needed.

Lemma 21. *Let M, N and O be three λ-terms of $\lambda^{\rightarrow\wedge\vee\forall\exists\bot}$ such that $M \rightarrow_\bot N$. Then, $M[x{:=}O] \rightarrow_\bot N[x{:=}O]$.*

Proof. By a straightforward induction on the derivation of $M \rightarrow_\bot N$. □

Lemma 22. *Let M, N and O be three λ-terms of $\lambda^{\rightarrow\wedge\vee\forall\exists\bot}$ such that $N \rightarrow_\bot O$. Then, $M[x{:=}N] \twoheadrightarrow_\bot M[x{:=}O]$.*

Proof. By a straightforward induction on the structure of M. □

We are now in a position of stating and proving the postponement lemma.

Lemma 23. *Let $R \in \{D, P, P_\exists\}$, and let L, M and N be three λ-terms of $\lambda^{\rightarrow\wedge\vee\forall\exists\bot}$ such that $L \rightarrow_\bot M \rightarrow_R N$. Then, there exists a λ-term O of $\lambda^{\rightarrow\wedge\vee\forall\exists\bot}$ such that $L \overset{+}{\rightarrow}_R O \twoheadrightarrow_\bot N$.*

Proof. The proof is done by induction on the structure of M, distinguishing between several subcases according to the way L reduces to M, and M reduces to N. Among these numerous subcases, the only ones that are not straightforward are these where L is a $\bot$-redex and where M, which is the contractum of L, is a R-redex whose contractum is N. In other words, the only non-trivial cases are these that result from a critical pair between the right-hand side of a rewriting rule of Definition 20 with the left-hand side of a rewriting rule of Definition 7, 14 or 15. All the right-hand sides of the rules of Definition 20 have the same shape, namely, $\bot(M')$. These right-hand sides match with the left-hand sides of two rules: Rule 7 of Definition 14, and Rules 7 of Definition 15. Consequently, there are fourteen subcases resulting from these critical pairs. We only give the first one, all the others being similar. Let L, M, and N be such that:

[3] In fact, it would have been the case if we had defined $\bot(M) : K$ to be $M : (\lambda x.\, x)$. But then, the infix operator ":" would no longer be strict in K, which implies the failure of Lemma 5 and, consequently, of Lemma 12.

$$
\begin{aligned}
L \;&=\; \bot(\mathbf{D}_{x,y}(M_1, M_2, M_3))\, L_1 \\
&\to_\bot\; \bot(\mathbf{D}_{x,y}(M_1, M_2, M_3)) \;=\; M \\
&\to_P\; \mathbf{D}_{x,y}(M_1, \bot(M_2), \bot(M_3)) \;=\; N
\end{aligned}
$$

Then, we have:

$$
\begin{aligned}
\bot(\mathbf{D}_{x,y}(M_1, M_2, M_3))\, L_1 &\to_P \mathbf{D}_{x,y}(M_1, \bot(M_2), \bot(M_3))\, L_1 \\
&\to_P \mathbf{D}_{x,y}(M_1, \bot(M_2)\, L_1, \bot(M_3)\, L_1) \\
&\to_\bot \mathbf{D}_{x,y}(M_1, \bot(M_2), \bot(M_3)\, L_1) \\
&\to_\bot \mathbf{D}_{x,y}(M_1, \bot(M_2), \bot(M_3))
\end{aligned}
$$

And we take $O = \mathbf{D}_{x,y}(M_1, \bot(M_2)\, L_1, \bot(M_3)\, L_1)$. □

The above lemma, together with Theorem 3 and Proposition 11, allows the last result of this paper to be established.

Theorem 4. $\lambda^{\to\wedge\vee\forall\exists\bot}$ *is strongly normalisable with respect to the reduction relation induced by the union of the detour-, permutation-, existential permutation-, and* $\bot$*-conversions.*

Proof. Suppose that there exists an infinite sequence of detour-, permutation-, existential permutation-, or $\bot$*-conversions. If this infinite sequence contains infinitely many detour-, permutation-, or existential permutation-conversion steps, one would construct, by applying repeatedly Lemma 23, an infinite sequence of such conversion steps. But this contradicts Theorem 3. Therefore, the infinite sequence may contain only a finite number of detour-, permutation-, or existential permutation-conversion steps. But then, it would contain an infinite sequence of consecutive* $\bot$*-conversion steps, which contradicts Proposition 11.* □

References

1. H.P. Barendregt. *The lambda calculus, its syntax and semantics.* North-Holland, revised edition, 1984.
2. H.B. Curry and R. Feys. *Combinatory Logic, Vol. I.* North-Holland, 1958.
3. Ph. de Groote. The conservation theorem revisited. In M. Bezem and J.F. Groote, editors, *Proceedings of the International Conference on Typed Lambda Calculi and Applications*, pages 163–178. Lecture Notes in Computer Science, 664, Springer Verlag, 1993.
4. Ph. de Groote. A simple calculus of exception handling. In M. Dezani and G. Plotkin, editors, *Second International Conference on Typed Lambda Calculi and Applications, TLCA'95*, volume 902 of *Lecture Notes in Computer Science*, pages 201–215. Springer Verlag, 1995.
5. G. Gentzen. *Recherches sur la déduction logique (Untersuchungen über das logische schliessen).* Presses Universitaires de France, 1955. Traduction et commentaire par R. Feys et J. Ladrière.
6. J.-Y. Girard. *Proof Theory and Logical Complexity.* Bibliopolis, 1987.
7. W.A. Howard. The formulae-as-types notion of construction. In J. P. Seldin and J. R. Hindley, editors, *to H. B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism*, pages 479–490. Academic Press, 1980.

8. A. Meyer and M. Wand. Continuation semantics in typed lambda-calculi (summary). In R. Parikh, editor, *Logics of Programs*, pages 219–224. Lecture Notes in Computer Science, 193, Springer Verlag, 1985.
9. G. Mints. Private communication, 1997.
10. G. D. Plotkin. Call-by-name, call-by-value and the λ-calculus. *Theoretical Computer Science*, 1:125–159, 1975.
11. D. Prawitz. *Natural Deduction, A Proof-Theoretical Study.* Almqvist & Wiksell, Stockholm, 1965.
12. D. Prawitz. Ideas and results in proof-theory. In J.E. Fenstad, editor, *Proceedings of the Second Scandinavian Logic Symposium*, pages 237–309. North-Holland, 1971.
13. A. Troelstra and D. van Dalen. *Constructivism in Mathematics*, volume I. North-Holland, 1988.
14. A. Troelstra and D. van Dalen. *Constructivism in Mathematics*, volume II. North-Holland, 1988.

A Proof of Propositions 2, 6 and 9

Variable

$$\dfrac{\dfrac{x : {\sim}{\sim}a \quad k : {\sim}a}{x\,k : o}}{\lambda k.\, x\,k : {\sim}{\sim}a}$$

Abstraction

$$\dfrac{\dfrac{k : {\sim}(\overline{\alpha} \to \overline{\beta}) \qquad \dfrac{\begin{matrix}[\,x : \overline{\alpha}\,] \\ \vdots \\ \overline{M} : \overline{\beta}\end{matrix}}{\lambda x.\, \overline{M} : \overline{\alpha} \to \overline{\beta}}}{k\,(\lambda x.\, \overline{M}) : o}}{\lambda k.\, k\,(\lambda x.\, \overline{M}) : {\sim}{\sim}(\overline{\alpha} \to \overline{\beta})}$$

Application

$$\dfrac{\dfrac{\overline{M} : {\sim}{\sim}(\overline{\alpha} \to \overline{\beta}) \qquad \dfrac{\dfrac{\dfrac{m : \overline{\alpha} \to \overline{\beta} \quad \overline{N} : \overline{\alpha}}{m\,\overline{N} : \overline{\beta}} \qquad k : {\sim}\beta^{\circ}}{m\,\overline{N}\,k : o}}{\lambda m.\, m\,\overline{N}\,k : {\sim}(\overline{\alpha} \to \overline{\beta})}}{\overline{M}\,(\lambda m.\, m\,\overline{N}\,k) : o}}{\lambda k.\, \overline{M}\,(\lambda m.\, m\,\overline{N}\,k) : \overline{\beta}}$$

Pairing

$$
\dfrac{k : \sim\sim(\overline{\alpha} \to \sim\overline{\beta}) \qquad \dfrac{\dfrac{\dfrac{\dfrac{p : \overline{\alpha} \to \sim\overline{\beta} \quad \overline{M} : \overline{\alpha}}{p\,\overline{M} : \sim\overline{\beta}} \qquad \overline{N} : \overline{\beta}}{p\,\overline{M}\,\overline{N} : o}}{\lambda p.\, p\,\overline{M}\,\overline{N} : \sim(\overline{\alpha} \to \sim\overline{\beta})}}{\dfrac{k\,(\lambda p.\, p\,\overline{M}\,\overline{N}) : o}{\lambda k.\, k\,(\lambda p.\, p\,\overline{M}\,\overline{N}) : \sim\sim\sim(\overline{\alpha} \to \sim\overline{\beta})}}
$$

Left projection

$$
\dfrac{\overline{M} : \sim\sim\sim(\overline{\alpha} \to \sim\overline{\beta}) \qquad \dfrac{\dfrac{p : \sim(\overline{\alpha} \to \sim\overline{\beta}) \quad \dfrac{\dfrac{\dfrac{i : \overline{\alpha} \quad k : \sim\alpha^{\circ}}{i\,k : o}}{\lambda j.\, i\,k : \sim\overline{\beta}}}{\lambda i.\, \lambda j.\, i\,k : \overline{\alpha} \to \sim\overline{\beta}}}{p\,(\lambda i.\, \lambda j.\, i\,k) : o}}{\lambda p.\, p\,(\lambda i.\, \lambda j.\, i\,k) : \sim\sim(\overline{\alpha} \to \sim\overline{\beta})}}{\dfrac{\overline{M}\,(\lambda p.\, p\,(\lambda i.\, \lambda j.\, i\,k)) : o}{\lambda k.\, \overline{M}\,(\lambda p.\, p\,(\lambda i.\, \lambda j.\, i\,k)) : \overline{\alpha}}}
$$

Right projection

$$
\dfrac{\overline{M} : \sim\sim\sim(\overline{\alpha} \to \sim\overline{\beta}) \qquad \dfrac{\dfrac{p : \sim(\overline{\alpha} \to \sim\overline{\beta}) \quad \dfrac{\dfrac{\dfrac{j : \overline{\beta} \quad k : \sim\beta^{\circ}}{j\,k : o}}{\lambda j.\, j\,k : \sim\overline{\beta}}}{\lambda i.\, \lambda j.\, j\,k : \overline{\alpha} \to \sim\overline{\beta}}}{p\,(\lambda i.\, \lambda j.\, j\,k) : o}}{\lambda p.\, p\,(\lambda i.\, \lambda j.\, j\,k) : \sim\sim(\overline{\alpha} \to \sim\overline{\beta})}}{\dfrac{\overline{M}\,(\lambda p.\, p\,(\lambda i.\, \lambda j.\, j\,k)) : o}{\lambda k.\, \overline{M}\,(\lambda p.\, p\,(\lambda i.\, \lambda j.\, j\,k)) : \overline{\beta}}}
$$

Left injection

$$
\dfrac{k : \sim(\sim\overline{\alpha} \to \sim\sim\overline{\beta}) \qquad \dfrac{\dfrac{\dfrac{i : \sim\overline{\alpha} \quad \overline{M} : \overline{\alpha}}{i\,\overline{M} : o}}{\lambda j.\, i\,\overline{M} : \sim\sim\overline{\beta}}}{\lambda i.\, \lambda j.\, i\,\overline{M} : \sim\overline{\alpha} \to \sim\sim\overline{\beta}}}{\dfrac{k\,(\lambda i.\, \lambda j.\, i\,\overline{M}) : o}{\lambda k.\, k\,(\lambda i.\, \lambda j.\, i\,\overline{M}) : \sim\sim(\sim\overline{\alpha} \to \sim\sim\overline{\beta})}}
$$

Right injection

$$\dfrac{\dfrac{k : \sim(\sim\overline{\alpha} \to \sim\sim\overline{\beta}) \qquad \dfrac{\dfrac{\dfrac{j : \sim\overline{\beta} \quad \overline{M} : \overline{\beta}}{j\,\overline{M} : o}}{\lambda j.\, j\,\overline{M} : \sim\sim\overline{\beta}}}{\lambda i.\, \lambda j.\, j\,\overline{M} : \sim\overline{\alpha} \to \sim\sim\overline{\beta}}}{k\,(\lambda i.\, \lambda j.\, j\,\overline{M}) : o}}{\lambda k.\, k\,(\lambda i.\, \lambda j.\, j\,\overline{M}) : \sim\sim(\sim\overline{\alpha} \to \sim\sim\overline{\beta})}$$

Case analysis

$$\dfrac{\dfrac{\overline{M} : \sim\sim(\sim\overline{\alpha} \to \sim\sim\overline{\beta}) \qquad \dfrac{\dfrac{\dfrac{m : \sim\overline{\alpha} \to \sim\sim\overline{\beta} \qquad \dfrac{\dfrac{\begin{matrix}[\, x : \overline{\alpha} \,] \\ \vdots \\ \overline{N} : \overline{\gamma}\end{matrix} \quad k : \sim\gamma^{\circ}}{\overline{N}\, k : o}}{\lambda x.\, \overline{N}\, k : \sim\overline{\alpha}}}{m\,(\lambda x.\, \overline{N}\, k) : \sim\sim\overline{\beta}} \qquad \dfrac{\dfrac{\begin{matrix}[\, y : \overline{\beta} \,] \\ \vdots \\ \overline{O} : \overline{\gamma}\end{matrix} \quad k : \sim\gamma^{\circ}}{\overline{O}\, k : o}}{\lambda y.\, \overline{O}\, k : \sim\overline{\beta}}}{m\,(\lambda x.\, \overline{N}\, k)\,(\lambda y.\, \overline{O}\, k) : o}}{\lambda m.\, m\,(\lambda x.\, \overline{N}\, k)\,(\lambda y.\, \overline{O}\, k) : \sim(\sim\overline{\alpha} \to \sim\sim\overline{\beta})}}{\overline{M}\,(\lambda m.\, m\,(\lambda x.\, \overline{N}\, k)\,(\lambda y.\, \overline{O}\, k)) : o}}{\lambda k.\, \overline{M}\,(\lambda m.\, m\,(\lambda x.\, \overline{N}\, k)\,(\lambda y.\, \overline{O}\, k)) : \overline{\gamma}}$$

∀-introduction

$$\dfrac{\dfrac{k : \sim(\iota \to \overline{\alpha}) \qquad \dfrac{\overline{M} : \overline{\alpha}}{\lambda \xi.\, \overline{M} : \iota \to \overline{\alpha}}}{k\,(\lambda \xi.\, \overline{M}) : o}}{\lambda k.\, k\,(\lambda \xi.\, \overline{M}) : \sim\sim(\iota \to \overline{\alpha})}$$

where ξ is a fresh variable of type ι.

∀-elimination

$$\dfrac{\dfrac{\overline{M} : \sim\sim(\iota \to \overline{\alpha}) \qquad \dfrac{\dfrac{\dfrac{m : \iota \to \overline{\alpha} \quad \star : \iota}{m\,\star : \overline{\alpha}} \qquad k : \sim\alpha^{\circ}}{m\,\star\, k : o}}{\lambda m.\, m\,\star\, k : \sim(\iota \to \overline{\alpha})}}{\overline{M}\,(\lambda m.\, m\,\star\, k) : o}}{\lambda k.\, \overline{M}\,(\lambda m.\, m\,\star\, k) : \overline{\alpha}}$$

∃-introduction

$$\dfrac{\dfrac{k : \sim\sim(\iota \to \sim\overline{\alpha}) \qquad \dfrac{\dfrac{\dfrac{\dfrac{a : \iota \to \sim\overline{\alpha} \quad \star : \iota}{a \star : \sim\overline{\alpha}} \qquad \overline{M} : \overline{\alpha}}{a \star \overline{M} : o}}{\lambda a.\, a \star \overline{M} : \sim(\iota \to \sim\overline{\alpha})}}{k\,(\lambda a.\, a \star \overline{M}) : o}}{\lambda k.\, k\,(\lambda a.\, a \star \overline{M}) : \sim\sim\sim(\iota \to \sim\overline{\alpha})}$$

∃-elimination

$$\dfrac{\dfrac{\overline{M} : \sim\sim\sim(\iota \to \sim\overline{\alpha}) \qquad \dfrac{\dfrac{m : \sim(\iota \to \sim\overline{\alpha}) \qquad \dfrac{\dfrac{\dfrac{\begin{matrix}[x : \overline{\alpha}] \\ \vdots \\ \overline{N} : \overline{\beta}\end{matrix} \qquad k : \sim\beta^{\circ}}{\overline{N}\,k : o}}{\lambda x.\, \overline{N}\,k : \sim\overline{\alpha}}}{\lambda \xi.\, \lambda x.\, \overline{N}\,k : \iota \to \sim\overline{\alpha}}}{m\,(\lambda \xi.\, \lambda x.\, \overline{N}\,k) : o}}{\lambda m.\, m\,(\lambda \xi.\, \lambda x.\, \overline{N}\,k) : \sim\sim(\iota \to \sim\overline{\alpha})}}{\overline{M}\,(\lambda m.\, m\,(\lambda \xi.\, \lambda x.\, \overline{N}\,k)) : o}}{\lambda k.\, \overline{M}\,(\lambda m.\, m\,(\lambda \xi.\, \lambda x.\, \overline{N}\,k)) : \overline{\beta}}$$

⊥-elimination

$$\dfrac{\dfrac{\overline{M} : (o \to o) \to o \qquad \dfrac{\dfrac{\dfrac{y : o}{\lambda y.\, y : o \to o}}{\lambda x.\, \lambda y.\, y : \sim\alpha^{\circ} \to (o \to o)} \qquad k : \sim\alpha^{\circ}}{(\lambda x.\, \lambda y.\, y)\,k : o \to o}}{\overline{M}\,((\lambda x.\, \lambda y.\, y)\,k) : o}}{\lambda k.\, \overline{M}\,((\lambda x.\, \lambda y.\, y)\,k) : \overline{\alpha}}$$

where x is of type $\sim\alpha^{\circ}$