

# ATOMIC POLYMORPHISM

FERNANDO FERREIRA AND GILDA FERREIRA

**Abstract.** It has been known for six years that the restriction of Girard’s polymorphic system  $\mathbf{F}$  to *atomic* universal instantiations interprets the full fragment of the intuitionistic propositional calculus. We firstly observe that Tait’s method of “convertibility” applies quite naturally to the proof of strong normalization of the restricted Girard system. We then show that each  $\beta$ -reduction step of the full intuitionistic propositional calculus translates into one or more  $\beta\eta$ -reduction steps in the restricted Girard system. As a consequence, we obtain a novel and perspicuous proof of the strong normalization property for the full intuitionistic propositional calculus. It is noticed that this novel proof bestows a crucial role to  $\eta$ -conversions.

**§1. Introduction.** In the early seventies, Jean-Yves Girard [5] and John Reynolds [13] introduced, independently, quantification over types in the  $\lambda$ -calculus and defined an attendant lambda system  $\mathbf{F}$  (a.k.a.  $\lambda\mathbf{2}$ ). The primitive version of this system is very elegant, with only two generators of types (formulas): implication and second-order universal quantification. Apart from the absence of first-order quantification (whose inclusion poses no particular problems, being essentially a matter of bearing with extra notation), the restriction to implication and second-order universal quantification does not limit the apparatus of logic. As it is well-known, Dag Prawitz [12] showed how to introduce the remaining logical connectives of falsum, conjunction and disjunction in second-order logic, thereby interpreting the *full* intuitionistic propositional calculus. System  $\mathbf{F}$  enjoys both the strong normalization and the Church-Rosser properties. The proof of the first of these properties is a somewhat delicate affair because second-order quantifications may instantiate arbitrary types (formulas). Due to this *impredicative* feature of  $\mathbf{F}$ , Girard’s normalization proof relies on strong forms of second-order comprehension via the ability to form sets of terms known as *reducibility candidates*.

---

*Key words and phrases.* Predicative polymorphism. Strong normalization. Natural deduction. Second-order  $\lambda$ -calculus.  $\eta$ -conversions.

Both authors acknowledge support of FCT-Fundação para a Ciência e a Tecnologia [project PTDC/MAT/104716/2008] and Centro de Matemática e Aplicações Fundamentais (Universidade de Lisboa). The second author is also grateful to FCT [grant SFRH/BPD/34527/2006] and Núcleo de Investigação em Matemática (Universidade Lusófona).

We study the restriction of system  $\mathbf{F}$  according to which the range of the type variables is constituted by the atomic types only. This system was introduced in [3] under the name of *atomic PSOL*<sup>i</sup>, where it was used as the basis for a commentary on the non-realistic cast of the philosophical position of predicativism. In that very paper it was observed that, even though only atomic instantiations are allowed, the restricted system – which we now denote by  $\mathbf{F}_{\text{at}}$  or  $\lambda_{\text{at}}$  – is still able to introduce the remaining logical connectives of falsum, conjunction and disjunction (via the very same definitions of Prawitz but, in our restricted case, also relying crucially on the so-called phenomenon of *instantiation overflow*, briefly recalled at the start of Section 4). The embedding of the full propositional calculus into  $\mathbf{F}_{\text{at}}$  was independently rediscovered by Tor Sandqvist [15].

In this paper, we give a proof of the strong normalization of  $\mathbf{F}_{\text{at}}$  with respect to  $\beta\eta$ -conversions and observe that the predicativity requirement of  $\mathbf{F}_{\text{at}}$  greatly simplifies the normalization argument. As an application, we present a new manner of proving strong normalization for the full intuitionistic propositional calculus with the standard  $\beta$ -conversions (using its embedding into  $\mathbf{F}_{\text{at}}$ ). Of course, the latter normalization result is known for a long time (see, for instance, [11]). However, the argument given in this paper seems to us specially perspicuous since it follows from a more general and natural result, viz. the above mentioned strong normalization of  $\mathbf{F}_{\text{at}}$ , and relies on some algorithms that may be of independent interest and worth analyzing in detail.

The paper is structured as follows. In the next section, we define and describe system  $\mathbf{F}_{\text{at}}$ . The restriction on the range of the type variables makes it easy to adapt Tait’s technique of *reducibility* (“convertibility,” in Tait’s original terminology [16]) in order to show that the system  $\mathbf{F}_{\text{at}}$  enjoys the property of strong normalization for  $\beta\eta$ -conversions. This is the subject of Section 3. Section 4 starts by briefly describing the embedding of the full intuitionistic propositional calculus into the system  $\mathbf{F}_{\text{at}}$ . The next order of business is to show that a  $\beta$ -conversion in the full language of the propositional intuitionistic calculus translates (via the embedding) into one or more  $\beta\eta$ -conversions in  $\mathbf{F}_{\text{at}}$ . Therefore, we are able to give a new proof of the strong normalization of the full intuitionistic propositional calculus. We finish the paper with some remarks.

**§2. The  $\mathbf{F}_{\text{at}}$  calculus: a predicative variant of system  $\mathbf{F}$ .** We present  $\mathbf{F}_{\text{at}}$  in the (operational)  $\lambda$ -calculus style. Later on, we also use a formulation in the natural deduction calculus. This dichotomy between the operational and deductive sides of the calculus (known as the Curry-Howard correspondence) is prevalent in the paper.

**DEFINITION 1.** *Types are constructed from atomic types (propositional constants  $P, Q, R, \dots$  and type variables  $X, Y, Z, \dots$ ) by means of two type-forming operations,  $\rightarrow$  and  $\forall$ , in the following way:*

- (i) *Atomic types are types.*

- (ii) If  $A$  and  $B$  are types then  $A \rightarrow B$  is a type.
- (iii) If  $A$  is a type and  $X$  is a type variable then  $\forall X.A$  is a type.

By regarding types as formulas, we have the usual definitions of *free* and *bound* (type) variables in a type. As usual, we can freely rename the bound variables in a type. We denote the set of free variables in  $A$  by  $\text{FV}(A)$ . Given a type  $A$ , a type variable  $X$  and an *atomic* type  $C$ , we write  $A[C/X]$  for the type obtained from  $A$  by substituting the free occurrences of  $X$  in  $A$  by  $C$  (without loss of generality, if  $C$  is itself a variable, we may assume that it is free for  $X$  in  $A$ ).

DEFINITION 2. Terms are generated by the following clauses:

- (i) For each type  $A$  there are countably infinite many assumption variables of type  $A$ ,  $x^A, y^A, z^A$ , etc. Assumption variables are terms.
- (ii) If  $t^{A \rightarrow B}$  and  $q^A$  are terms of types  $A \rightarrow B$  and  $A$ , respectively, then  $(t^{A \rightarrow B} q^A)^B$  is a term of type  $B$ .
- (iii) If  $t^B$  is a term of type  $B$  and  $x^A$  is an assumption variable of type  $A$ , then  $(\lambda x^A. t^B)^{A \rightarrow B}$  is a term of type  $A \rightarrow B$ . A term of this form is called an *arrow abstraction*.
- (iv) If  $t^{\forall X.A}$  is a term of type  $\forall X.A$  and  $C$  in an atomic type, then  $(t^{\forall X.A} C)^{A[C/X]}$  is a term of type  $A[C/X]$ .
- (v) If  $t^A$  is a term of type  $A$  and the type variable  $X$  is not free in the type of any free assumption variable of  $t^A$ , then  $(\Lambda X. t^A)^{\forall X.A}$  is a term of type  $\forall X.A$ . A term of this form is called a *universal abstraction*.

When it is clear from the context, we sometimes omit the (type) superscript in  $t^A$ , denoting the term simply by  $t$ . The difference between Girard's system  $\mathbf{F}$  and the restriction  $\mathbf{F}_{\text{at}}$  lies in the fourth clause above:  $\mathbf{F}$  allows the construction of terms of the form  $(tD)^{A[D/X]}$ , with  $D$  any type whatsoever;  $\mathbf{F}_{\text{at}}$  only permits second-order application to atomic types. This difference explains the impredicativity of system  $\mathbf{F}$  (e.g.,  $D$  can be  $\forall X.A$  itself), against the predicativity of  $\mathbf{F}_{\text{at}}$ . Note, also, that in Girard's system there is no sensible notion of subformula because universal types  $\forall X.A$  allow instantiations  $A[D/X]$  for any type  $D$ , however complex. In contrast, there is a natural definition of subformula within the types of  $\mathbf{F}_{\text{at}}$ : we say that the immediate subformulas of  $\forall X.A$  are the formulas of the form  $A[C/X]$ , where  $C$  is an atomic type (free for  $X$  in  $A$ ). The notion of subformula provides attendant notions of definition and proof by the complexity of formulas. This feature will play a crucial role in the normalization proof below.

DEFINITION 3. The set of free variables in a term  $t$ , denoted by  $\text{FV}(t)$ , is defined inductively by:

$$\begin{aligned} \text{FV}(x^A) &::= \{x^A\} \cup \text{FV}(A) \\ \text{FV}(tq) &::= \text{FV}(t) \cup \text{FV}(q) \\ \text{FV}(\lambda x^A. t) &::= (\text{FV}(t) \setminus \{x^A\}) \cup \text{FV}(A) \\ \text{FV}(tC) &::= \text{FV}(t) \cup \text{FV}(C) \end{aligned}$$

$$\text{FV}(\Lambda X.t) := \text{FV}(t) \setminus \{X\}$$

Type and assumption variables not free in  $t$  are called bound variables in  $t$ .

When necessary, we may assume that the sets of free and bound variables in a term are disjoint (all expressions are considered modulo a renaming of the bound variables). The substitution of a free assumption variable  $x^A$  in a term  $t$  by a term  $s^A$  is done in the expected way (see [17]) and denoted by  $t[s/x]$ .

**DEFINITION 4.** Given a term  $t$  of type  $A$  and  $C$  an atomic type, we define a new term  $t[C/X]$  of type  $A[C/X]$  inductively as follows:

$$\begin{aligned} x^A[C/X] &:= x^{A[C/X]} \\ (qr)[C/X] &:= q[C/X]r[C/X] \\ (\lambda x^A.q)[C/X] &:= \lambda x^{A[C/X]}.q[C/X] \\ (\Lambda X.q)[C/X] &:= \Lambda X.q \\ (\Lambda Y.q)[C/X] &:= \Lambda Y.(q[C/X]), \text{ for } Y \neq X, \text{ w.l.o.g. } C \neq Y \\ (qD)[C/X] &:= q[C/X]D[C/X], \text{ with } D \text{ an atomic type.} \end{aligned}$$

Again we assume that there is no clash of variables, i.e., in case  $C$  is a type variable, we assume that it is free for  $X$  in  $q$ ,  $r$  and  $A$ . It can easily be shown, by induction on the complexity of the term  $t$ , that  $t[C/X]$  above is well-defined. Free type variables can be regarded as parameters, in the sense that we can substitute them by propositional constants. After such a substitution, a term does not have free type variables. Now, consider a free assumption variable  $x^A$  in such a term (after the substitution is effected). Does it have a determinate type? In other words, is  $A$  a *sentence*? The answer is yes, and this is due to the important restriction (v) of Definition 2. This means that the type of a free assumption variable in a term  $t$  depends only on the free type variables of  $t$ .

We need also the concept of *simultaneous* substitution. We skip its definition and use for this kind of substitution the following notations:  $t[s_1/x_1, \dots, s_n/x_n]$  and  $t[C_1/X_1, \dots, C_n/X_n]$ .

In analogy with the system  $\mathbf{F}$ , we have two  $\beta$ -conversions: one for implication, the *arrow  $\beta$ -conversion*, and the other for second-order universal quantification, the *universal  $\beta$ -conversion*. They are, respectively,

$$\begin{aligned} (\lambda x.t)s &\rightsquigarrow t[s/x] \\ (\Lambda X.t)C &\rightsquigarrow t[C/X], \end{aligned}$$

where the left hand side of a conversion is called its *redex* and the right hand side its *contractum*. It is clear that the previous conversions are well defined, originating contractums in  $\mathbf{F}_{\text{at}}$  of the same type of the redexes. We also need in the sequel the so-called  $\eta$ -conversions:

$$\begin{aligned} \lambda x.(tx) &\rightsquigarrow t, \text{ with } x \notin \text{FV}(t) \\ \Lambda X.(tX) &\rightsquigarrow t, \text{ with } X \notin \text{FV}(t). \end{aligned}$$

The first one is the *arrow  $\eta$ -conversion*, the second is the *universal  $\eta$ -conversion*. Again they are well defined in  $\mathbf{F}_{\text{at}}$  and terms on the left hand side are called *redexes* and on the right hand side are called *contractums*.

DEFINITION 5. A term  $t$  reduces to a term  $q$  (and we write  $t \geq q$ ) if there is a sequence of  $\beta\eta$ -conversions from  $t$  to  $q$ , i.e., a sequence  $t \equiv u_0, u_1, \dots, u_n \equiv q$ , such that for  $i = 0, 1, \dots, n-1$ ,  $u_{i+1}$  is obtained from  $u_i$  by replacing a redex by its contractum. A term is normal if it has no redexes and so we can not apply any conversion any longer. A term  $t$  is strongly normalizable if all the reduction sequences starting with  $t$  have finite length.

When a term  $t$  reduces to a term  $q$  in one step, we write  $t >_1 q$ . If  $t$  is strongly normalizable, we denote by  $\mu(t)$  a number that bounds the length of every reduction sequence beginning with  $t$ . Such bound exists because the reduction tree of a term is finitely branching and, hence, we can apply König's lemma.

By the Curry-Howard correspondence, we can present the system  $\mathbf{F}_{\text{at}}$  as a natural deduction calculus. The formulas of  $\mathbf{F}_{\text{at}}$  are the types: formulas generated by only two primitive logic connectives, viz the implication and the second-order universal quantifier. Natural deductions in intuitionistic logic correspond to terms of  $\mathbf{F}_{\text{at}}$ . As usual, the introduction rules are

$$\frac{[A] \quad \dots \quad \dots \quad B}{A \rightarrow B} \rightarrow\text{I} \qquad \frac{\dots \quad \dots \quad A}{\forall X.A} \forall\text{I}$$

where, in the second rule,  $X$  does not occur free in any undischarged hypothesis. The elimination rules are

$$\frac{\dots \quad \dots \quad A \rightarrow B \quad A}{B} \rightarrow\text{E} \qquad \frac{\dots \quad \dots \quad \forall X.A}{A[C/X]} \forall\text{E}$$

with  $C$  an *atomic* formula, free for  $X$  in  $A$ . Again, it is the restriction on  $C$  that distinguishes  $\mathbf{F}_{\text{at}}$  from  $\mathbf{F}$ .

In the natural deduction formulation, the arrow  $\beta$ -conversions and the universal  $\beta$ -conversions have the following well-known forms (respectively):

$$\frac{[A] \quad \dots \quad \dots \quad B \quad A}{A \rightarrow B \quad B} \rightsquigarrow \frac{\dots \quad \dots \quad A}{A} \quad \dots \quad A$$

$$\frac{\dots \quad \dots \quad A}{\forall X.A} \rightsquigarrow \frac{\dots \quad \dots \quad A}{A[C/X]} \quad \dots \quad A[C/X]$$

where  $C$  is an *atomic* formula, free for  $X$  in  $A$ . The arrow  $\eta$ -conversions and the universal  $\eta$ -conversions are expressed, respectively, by:

$$\begin{array}{ccc} \begin{array}{c} \vdots \\ \frac{A \rightarrow B \quad A}{B} \\ \frac{B}{A \rightarrow B} \end{array} & \rightsquigarrow & \begin{array}{c} \vdots \\ A \rightarrow B \end{array} \\ \\ \begin{array}{c} \vdots \\ \frac{\forall X.A}{A} \\ \forall X.A \end{array} & \rightsquigarrow & \begin{array}{c} \vdots \\ \forall X.A \end{array} \end{array}$$

**§3. The strong normalization proof of  $\mathbf{F}_{\text{at}}$ .** In this section, we adapt/extend Tait's method in order to show that  $\mathbf{F}_{\text{at}}$  is strongly normalizable.

**DEFINITION 6.** We say that the term  $t^A$  is reducible if  $t \in \text{Red}_A$ , with  $\text{Red}_A$  defined by induction on the complexity of the type (formula)  $A$  as follows:

- For  $C$  an atomic type,  $t \in \text{Red}_C := t$  is strongly normalizable.
- $t \in \text{Red}_{A \rightarrow B} :=$  for all  $q$ , if  $q \in \text{Red}_A$  then  $tq \in \text{Red}_B$ .
- $t \in \text{Red}_{\forall X.A} :=$  for all atomic types  $C$ ,  $tC \in \text{Red}_{A[C/X]}$ .

Notice that in system  $\mathbf{F}$  it wouldn't make sense to define a third clause as follows:  $t \in \text{Red}_{\forall X.A} :=$  for all types  $C$ ,  $tC \in \text{Red}_{A[C/X]}$ . In effect, the type  $A[C/X]$  could be of arbitrary high complexity, and the definition by induction would fail.

Let us call *neutral terms* the terms of the form  $x$ ,  $tq$  or  $tC$  (i.e., terms which are not abstractions).

**LEMMA 1.**  $\text{Red}_A$  satisfies the following three conditions:

- (CR 1) If  $t \in \text{Red}_A$  then  $t$  is strongly normalizable.
- (CR 2) If  $t \in \text{Red}_A$  and  $t \geq t'$  then  $t' \in \text{Red}_A$ .
- (CR 3) If  $t$  is neutral and  $t' \in \text{Red}_A$  for all  $t'$  such that  $t >_1 t'$ , then  $t \in \text{Red}_A$ .

As a special case of CR 3, we have in particular:

- (CR 4) If  $t^A$  is neutral and normal then  $t \in \text{Red}_A$ .

**PROOF.** The proof is by induction on the complexity of the type (formula)  $A$ . For the atomic and arrow types, the proof follows exactly as in [6], Section 6.2. We just have to analyze the type  $A := \forall X.B$ .

Take  $t \in \text{Red}_{\forall X.B}$  in order to prove that  $t$  is strongly normalizable. By definition of  $\text{Red}_{\forall X.B}$ ,  $tX \in \text{Red}_B$ . By induction hypothesis,  $\text{Red}_B$  satisfies CR 1, so  $tX$  is strongly normalizable. Clearly, each reduction sequence for  $t$  gives rise to a reduction sequence for  $tX$  by applying each term of the sequence to  $X$ . Hence,  $t$  is strongly normalizable.

In order to prove CR 2, let  $t \in \text{Red}_{\forall X.B}$  and  $t \geq t'$ . We want to show that  $t' \in \text{Red}_{\forall X.B}$ . Let  $C$  be an atomic type. We know that  $tC \in \text{Red}_{B[C/X]}$ . Since  $tC \geq t'C$ , by induction hypothesis (CR 2, for  $\text{Red}_{B[C/X]}$ ), we have  $t'C \in \text{Red}_{B[C/X]}$ .

For condition CR 3, take  $t^{yX.B}$  as in the hypothesis. We want to prove that  $t \in \text{Red}_{yX.B}$ . Let  $C$  be an atomic type. Note that  $tC$  is neutral and that the redexes in  $tC$  occur in  $t$  (since  $t$  is neutral and, hence, not a universal abstraction). Therefore, any single reduction step from  $tC$  has the form  $t'C$  with  $t >_1 t'$ . By hypothesis  $t' \in \text{Red}_{yX.B}$ , and we get  $t'C \in \text{Red}_{B[C/X]}$ . Applying now the induction hypothesis (CR 3 for  $\text{Red}_{B[C/X]}$ ), we conclude that  $tC \in \text{Red}_{B[C/X]}$ .  $\dashv$

LEMMA 2. *If for all reducible  $q$  of type  $A$ ,  $t[q/x^A]$  is reducible, then so is  $\lambda x.t$ .*

PROOF. By definition, we need to show that, for every reducible  $u$  (of type  $A$ ),  $(\lambda x.t)u$  is reducible. The proof is by induction on  $\mu(t) + \mu(u)$ . By CR 3, we prove that every one step reduction from  $(\lambda x.t)u$  is reducible. The possible one step reducts are the following:  $t[u/x]$ ,  $(\lambda x.t')u$ ,  $(\lambda x.t)u'$  and  $su$ , where in the second case  $t >_1 t'$ , in the third case  $u >_1 u'$  and in the last case  $t$  is  $sx$  (for  $t'$ ,  $u'$  and  $s$  appropriate terms, the latter one without free occurrences of  $x$ ). The first three cases were studied in [6], Subsection 6.3.2. The last case reduces to the first since  $su \equiv (sx)[u/x] \equiv t[u/x]$ .  $\dashv$

We need to have an analogous result for universal abstraction.

LEMMA 3. *If for all atomic types  $C$ ,  $t[C/X]$  is reducible and  $X$  is not free in the type of a free assumption variable of  $t$ , then  $\Lambda X.t$  is reducible.*

PROOF. We prove that, for all terms  $t$  satisfying the conditions of the lemma, we have: For all atomic types  $C$ ,  $(\Lambda X.t)C$  is reducible. The proof is by induction on  $\mu(t)$ . (Indeed,  $t$  is strongly normalizable because  $t$  is  $t[X/X]$  which, by supposition, is reducible and hence, by CR 1 is strongly normalizable.)

Fix  $C$  an atomic type. Since the term  $(\Lambda X.t)C$  is neutral, we only have to show that the one-step reducts from this term are reducible (thanks to CR 3). Such a one step reduct must be of the form  $t[C/X]$ ,  $(\Lambda X.t')C$  with  $t >_1 t'$  or  $sC$ , when  $t$  is  $sX$  and  $X$  does not occur free in  $s$ . In the first case, reducibility follows from the hypothesis. In the second case, note that  $\mu(t') < \mu(t)$  and that, for all atomic types  $D$ ,  $t'[D/X]$  is reducible (the last assertion uses CR 2). So, by induction hypothesis,  $(\Lambda X.t')C$  is reducible. The third case reduces to the first since  $sC \equiv (sX)[C/X] \equiv t[C/X]$ .  $\dashv$

PROPOSITION 1. *Let  $t$  be any term (not assumed to be reducible) and suppose that all the free assumption variables of  $t$  are among  $x_1, \dots, x_n$ , of types  $A_1, \dots, A_n$  (respectively). Suppose, in addition, that all the free type variables of  $t$  are among  $X_1, \dots, X_m$ . If  $C_1, \dots, C_m$  are atomic types and  $q_1, \dots, q_n$  are reducible terms of types  $A_1[C_1/X_1, \dots, C_m/X_m], \dots, A_n[C_1/X_1, \dots, C_m/X_m]$  then the term  $t[C_1/X_1, \dots, C_m/X_m][q_1/x_1, \dots, q_n/x_n]$  is reducible.*

PROOF. The proof is by induction on the complexity of the term  $t$ . For  $t$  of the form  $x^A$ ,  $ps$  or  $\lambda y.p$  the argument is an easy adaptation of the one presented in

[6], Subsection 6.3.3 (using Definition 4 and Lemma 2 above). The new cases are universal application and type abstraction.

Let  $t := (p^{Y.B}D)^{B[D/Y]}$ , with  $D$  an atomic type. Take  $Z$  a type variable such that  $\text{FV}(D) \subseteq \{Z\}$ , i.e. we are assuming that if  $D$  is a type variable then it is the variable  $Z$ . We want to show that the term  $(pD)[\bar{C}/\bar{X}, P/Z][\bar{q}/\bar{x}]$  is reducible with  $\bar{C}$  and  $P$  atomic types and  $\bar{q}$  a sequence of reducible terms of appropriate types. By induction hypothesis we know that  $p[\bar{C}/\bar{X}, P/Z][\bar{q}/\bar{x}]$  is reducible of type  $\forall Y.B[\bar{C}/\bar{X}, P/Z]$ . By definition of reducibility, we infer that the term  $(p[\bar{C}/\bar{X}, P/Z][\bar{q}/\bar{x}])D[P/Z]$  is reducible. But

$$(pD)[\bar{C}/\bar{X}, P/Z][\bar{q}/\bar{x}] \equiv (p[\bar{C}/\bar{X}, P/Z][\bar{q}/\bar{x}])D[P/Z],$$

as wanted.

For  $t := \Lambda Y.p$ , we want to prove that  $(\Lambda Y.p)[\bar{C}/\bar{X}][\bar{q}/\bar{x}]$  is reducible. By definition of substitution, this term is  $\Lambda Y.(p[\bar{C}/\bar{X}][\bar{q}/\bar{x}])$ . By Lemma 3, it is enough to show that, for all atomic types  $D$ ,  $(p[\bar{C}/\bar{X}][\bar{q}/\bar{x}])[D/Y]$  is reducible. Well, the latter term is just  $p[\bar{C}/\bar{X}, D/Y][\bar{q}/\bar{x}]$ , and this one is reducible by induction hypothesis.  $\dashv$

**THEOREM 1.** *All terms of  $\mathbf{F}_{\text{at}}$  are reducible.*

**PROOF.** Let  $t[X_1, \dots, X_m][x_1, \dots, x_n]$  be an arbitrary term, where all the free variables are among the ones presented. Since each  $x_i$ , for  $i := 1, \dots, n$ , is neutral and normal, we know by CR 4 it is reducible.  $X_j$  for  $j := 1, \dots, m$  are type variables, so they are atomic types. By Proposition 1, we have that  $t[X_1, \dots, X_m][x_1, \dots, x_n]$  is reducible.  $\dashv$

The following result is now immediate, using condition CR 1.

**COROLLARY 1.** *The system  $\mathbf{F}_{\text{at}}$  with  $\beta\eta$ -conversions is strongly normalizable.*

**§4. The reduction into the  $\mathbf{F}_{\text{at}}$  calculus.** It was shown in [3, 4] that the full intuitionistic propositional calculus can be embedded into  $\mathbf{F}_{\text{at}}$  as follows:

$$\begin{aligned} \perp &::= \forall X.X \\ A \wedge B &::= \forall X((A \rightarrow (B \rightarrow X)) \rightarrow X) \\ A \vee B &::= \forall X((A \rightarrow X) \rightarrow ((B \rightarrow X) \rightarrow X)), \end{aligned}$$

where  $X$  is a second-order variable which does not occur in  $A$  or  $B$ . This embedding relies on the property of *instantiation overflow*: From formulas of the form above, it is possible to deduce in  $\mathbf{F}_{\text{at}}$  (respectively)

$$\begin{aligned} C \\ (A \rightarrow (B \rightarrow C)) \rightarrow C \\ (A \rightarrow C) \rightarrow ((B \rightarrow C) \rightarrow C), \end{aligned}$$

for *any* (not necessarily atomic) formula  $C$ . In other words, for formulas of the form above (i.e., of the form  $\perp$ ,  $A \wedge B$  or  $A \vee B$ ), instantiation overflow ensures that universal instantiation is no longer restricted to atomic formulas. (Notice that



unrestricted universal instantiations are always automatic in the impredicative system  $\mathbf{F}$ .) The proof of instantiation overflow is by induction on the complexity of  $C$  and provides an algorithmic method for obtaining the three kinds of deductions above.

We are not merely going to use instantiation overflow but also the *algorithmic manner in which it is obtained*. In order to make the paper reasonably self-contained, we present here the algorithm for disjunction. I.e., we show how to deduce in  $\mathbf{F}_{\text{at}}$  the formula  $(A \rightarrow C) \rightarrow ((B \rightarrow C) \rightarrow C)$ , for *arbitrary*  $C$ , from  $A \vee B$  as defined above. For  $C$  atomic, there is nothing to argue. We just have to analyze the cases in which  $C$  is  $D_1 \rightarrow D_2$  or  $\forall XD$  admitting (by induction hypothesis) that instantiation overflow is available for  $D_1$  and  $D_2$ , or  $D$  (respectively). For  $C := D_1 \rightarrow D_2$ , we have:

$$\frac{\frac{\frac{\forall X((A \rightarrow X) \rightarrow ((B \rightarrow X) \rightarrow X))}{(A \rightarrow D_2) \rightarrow ((B \rightarrow D_2) \rightarrow D_2)} \text{ I.H.}}{(B \rightarrow D_2) \rightarrow D_2}}{\frac{\frac{\frac{[A \rightarrow (D_1 \rightarrow D_2)] \quad [A]}{D_1 \rightarrow D_2} \quad [D_1]}{D_2}}{A \rightarrow D_2}}{(B \rightarrow (D_1 \rightarrow D_2)) \rightarrow (D_1 \rightarrow D_2)}}{\frac{D_2}{D_1 \rightarrow D_2}} \mathcal{D}}{\frac{(B \rightarrow (D_1 \rightarrow D_2)) \rightarrow (D_1 \rightarrow D_2)}{(A \rightarrow (D_1 \rightarrow D_2)) \rightarrow ((B \rightarrow (D_1 \rightarrow D_2)) \rightarrow (D_1 \rightarrow D_2))}}{\mathcal{D}}$$

where  $\mathcal{D}$  is the deduction

$$\frac{\frac{[B \rightarrow (D_1 \rightarrow D_2)] \quad [B]}{D_1 \rightarrow D_2} \quad [D_1]}{D_2}}{B \rightarrow D_2}$$

For  $C := \forall XD$  we have:

$$\frac{\frac{\frac{\frac{\forall X((A \rightarrow X) \rightarrow ((B \rightarrow X) \rightarrow X))}{(A \rightarrow D) \rightarrow ((B \rightarrow D) \rightarrow D)} \text{ I.H.}}{(B \rightarrow D) \rightarrow D}}{\frac{D}{\forall XD}}}{(B \rightarrow \forall XD) \rightarrow \forall XD}}{\frac{\frac{[A \rightarrow \forall XD] \quad [A]}{\forall XD} \quad [B]}{A \rightarrow D} \quad \frac{[B \rightarrow \forall XD] \quad [B]}{\forall XD}}{B \rightarrow D}}{\frac{D}{\forall XD}} \mathcal{D}}{\frac{(B \rightarrow \forall XD) \rightarrow \forall XD}{(A \rightarrow \forall XD) \rightarrow ((B \rightarrow \forall XD) \rightarrow \forall XD)}}{\mathcal{D}}$$

We refer to the above procedure as the *canonical* way of disclosing the portion of the proof hidden when using an instantiation overflow. In the proof trees below (as above), the instantiation overflow, with the accompanying canonical hidden portion, is signaled by a double horizontal line.

Having the property of instantiation overflow in place, full intuitionistic logic embeds into  $\mathbf{F}_{\text{at}}$  following Prawitz's recipe of [12] to the latter. This embedding provides a *canonical* way of translating proofs, rule-by-rule. The reader is

referred to [3, 4] for an explicit account. Nevertheless, since the details of the embedding are needed in the arguments below, we describe here the translations of the natural deduction rules for disjunction. The introduction rule  $\frac{A}{A \vee B}$  is translated into  $\mathbf{F}_{\text{at}}$  by

$$\frac{\frac{\frac{A \quad [A \rightarrow X]}{X}}{(B \rightarrow X) \rightarrow X}}{(A \rightarrow X) \rightarrow ((B \rightarrow X) \rightarrow X)}}{\forall X[(A \rightarrow X) \rightarrow ((B \rightarrow X) \rightarrow X)]}$$

and the elimination rule

$$\frac{\begin{array}{c} \vdots \\ \vdots \\ A \vee B \end{array} \quad \begin{array}{c} [A] \\ \vdots \\ \vdots \\ C \end{array} \quad \begin{array}{c} [B] \\ \vdots \\ \vdots \\ C \end{array}}{C}$$

has the following translation into  $\mathbf{F}_{\text{at}}$

$$\frac{\frac{\frac{\frac{\vdots}{\forall X((A \rightarrow X) \rightarrow ((B \rightarrow X) \rightarrow X))}}{(A \rightarrow C) \rightarrow ((B \rightarrow C) \rightarrow C)}}{(B \rightarrow C) \rightarrow C} \quad \frac{\frac{[A] \quad \vdots \quad \vdots \quad C}{A \rightarrow C}}{C} \quad \frac{[B] \quad \vdots \quad \vdots \quad C}{B \rightarrow C}}{C}$$

where the double line hides the instantiation overflow discussed before. For ease of notation, in the translations above, we ignored the translations of  $A$ ,  $B$  and  $C$ .

From now on, all the translations of proofs from the intuitionistic propositional calculus into  $\mathbf{F}_{\text{at}}$  are assumed to be obtained in this canonical way and are called *canonical translations*.

The main purpose of this section is to give a proof of the strong normalization for the full intuitionistic propositional calculus (with the standard  $\beta$ -conversions) via the strong normalization for  $\mathbf{F}_{\text{at}}$ . Let us start with a technical result:

LEMMA 4. *A derivation of the form*

$$\frac{\frac{\frac{\frac{\frac{\vdots}{A} \quad [A \rightarrow X]}{X}}{(B \rightarrow X) \rightarrow X}}{(A \rightarrow X) \rightarrow ((B \rightarrow X) \rightarrow X)}}{\forall X[(A \rightarrow X) \rightarrow ((B \rightarrow X) \rightarrow X)]} \quad \frac{[A] \quad \vdots \quad \vdots \quad C}{A \rightarrow C} \quad \frac{[B] \quad \vdots \quad \vdots \quad C}{B \rightarrow C}}{(B \rightarrow C) \rightarrow C} \quad C$$

$\vdots$   
 $\vdots$   
 reduces to  $\frac{A}{\vdots}$  using  $\beta\eta$ -conversions.  
 $\vdots$   
 $C$

**PROOF.** The proof is by induction on the complexity of the formula  $C$ . If  $C$  is an atomic formula only  $\beta$ -conversions are needed, more precisely four  $\beta$ -conversions. Using a universal  $\beta$ -conversion, we get

$$\begin{array}{c}
 \vdots \\
 \frac{A \quad [A \rightarrow C]}{C} \quad \frac{[A]}{\vdots} \quad \frac{[B]}{\vdots} \\
 \frac{(B \rightarrow C) \rightarrow C}{(A \rightarrow C) \rightarrow ((B \rightarrow C) \rightarrow C)} \quad \frac{A \rightarrow C}{A \rightarrow C} \quad \frac{C}{B \rightarrow C} \\
 \frac{(B \rightarrow C) \rightarrow C}{C}
 \end{array}$$

because, with  $C$  an atomic formula, the double line in the derivation is in fact the application of a single inference rule. With two more arrow  $\beta$ -conversions we get

$$\begin{array}{c}
 [A] \\
 \vdots \\
 \frac{C}{A \rightarrow C} \quad \text{and with one more arrow } \beta\text{-conversion we obtain} \quad \frac{A}{\vdots} \\
 \frac{A}{C} \quad \frac{A \rightarrow C}{C} \quad \frac{C}{C}
 \end{array}$$

If  $C := D_1 \rightarrow D_2$  and we disclose a bit of the portion hidden in the double line (see the proof of disjunction overflow for the arrow case), we get

$$\begin{array}{c}
 \vdots \\
 \frac{A \quad [A \rightarrow X]}{X} \\
 \frac{(B \rightarrow X) \rightarrow X}{(A \rightarrow X) \rightarrow ((B \rightarrow X) \rightarrow X)} \quad \frac{[A \rightarrow (D_1 \rightarrow D_2)] \quad [A]}{D_1 \rightarrow D_2} \quad [D_1] \\
 \frac{\forall X[(A \rightarrow X) \rightarrow ((B \rightarrow X) \rightarrow X)]}{(A \rightarrow D_2) \rightarrow ((B \rightarrow D_2) \rightarrow D_2)} \quad \frac{D_2}{A \rightarrow D_2} \\
 \frac{(B \rightarrow D_2) \rightarrow D_2}{(B \rightarrow D_2) \rightarrow D_2} \quad \mathcal{D} \\
 \frac{D_2}{D_1 \rightarrow D_2} \\
 \frac{(B \rightarrow (D_1 \rightarrow D_2)) \rightarrow (D_1 \rightarrow D_2)}{(A \rightarrow (D_1 \rightarrow D_2)) \rightarrow ((B \rightarrow (D_1 \rightarrow D_2)) \rightarrow (D_1 \rightarrow D_2))}
 \end{array}$$

above the formula  $(A \rightarrow C) \rightarrow ((B \rightarrow C) \rightarrow C)$ , where  $\mathcal{D}$  is the deduction

$$\begin{array}{c}
 \frac{[B \rightarrow (D_1 \rightarrow D_2)] \quad [B]}{D_1 \rightarrow D_2} \quad [D_1] \\
 \frac{D_2}{B \rightarrow D_2}
 \end{array}$$

By induction hypothesis, the derivation reduces to

$$\begin{array}{c}
 \vdots \\
 \vdots \\
 \frac{A \quad [A \rightarrow (D_1 \rightarrow D_2)]}{D_1 \rightarrow D_2} \quad [D_1] \\
 \frac{D_2}{D_1 \rightarrow D_2} \\
 \frac{(B \rightarrow (D_1 \rightarrow D_2)) \rightarrow (D_1 \rightarrow D_2)}{(A \rightarrow (D_1 \rightarrow D_2)) \rightarrow ((B \rightarrow (D_1 \rightarrow D_2)) \rightarrow (D_1 \rightarrow D_2))} \\
 \frac{(A \rightarrow (D_1 \rightarrow D_2)) \rightarrow ((B \rightarrow (D_1 \rightarrow D_2)) \rightarrow (D_1 \rightarrow D_2)) \quad (B \rightarrow (D_1 \rightarrow D_2)) \rightarrow (D_1 \rightarrow D_2)}{D_1 \rightarrow D_2} \\
 \end{array}
 \quad
 \frac{
 \begin{array}{c}
 [A] \\
 \vdots \\
 \vdots \\
 D_1 \rightarrow D_2 \\
 \hline
 A \rightarrow (D_1 \rightarrow D_2) \\
 \hline
 D_1 \rightarrow D_2 \\
 \hline
 B \rightarrow (D_1 \rightarrow D_2) \\
 \hline
 [D_1]
 \end{array}
 }{D_1 \rightarrow D_2}$$

by means of  $\beta\eta$ -conversions. Note that we have only changed the portion of derivation above  $D_2$ .

Applying two arrow  $\beta$ -conversions we obtain

$$\frac{
 \begin{array}{c}
 [A] \\
 \vdots \\
 \vdots \\
 D_1 \rightarrow D_2 \\
 \hline
 A \rightarrow (D_1 \rightarrow D_2) \\
 \hline
 D_1 \rightarrow D_2 \\
 \hline
 D_2 \\
 \hline
 D_1 \rightarrow D_2 \\
 \hline
 [D_1]
 \end{array}
 }{D_1 \rightarrow D_2}$$

and with one more arrow  $\beta$ -conversion we get

$$\frac{
 \begin{array}{c}
 A \\
 \vdots \\
 \vdots \\
 D_1 \rightarrow D_2 \quad [D_1] \\
 \hline
 D_2 \\
 \hline
 D_1 \rightarrow D_2
 \end{array}
 }{D_1 \rightarrow D_2}$$

Finally, with one arrow  $\eta$ -conversion the derivation reduces to

$$\begin{array}{c}
 A \\
 \vdots \\
 \vdots \\
 D_1 \rightarrow D_2
 \end{array}$$

If  $C := \forall Y.D$  and we disclose a bit of the portion hidden in the double line (see the proof of disjunction overflow for the universal case), we get

$$\frac{
 \frac{
 \frac{
 \frac{
 \frac{
 \frac{
 \frac{
 \frac{
 \frac{A \quad [A \rightarrow X]}{X} \\
 (B \rightarrow X) \rightarrow X \\
 \hline
 (A \rightarrow X) \rightarrow ((B \rightarrow X) \rightarrow X) \\
 \hline
 \forall X[(A \rightarrow X) \rightarrow ((B \rightarrow X) \rightarrow X)] \\
 \hline
 (A \rightarrow D) \rightarrow ((B \rightarrow D) \rightarrow D) \\
 \hline
 (B \rightarrow D) \rightarrow D
 \end{array}
 \quad
 \frac{
 \frac{
 \frac{[A \rightarrow \forall Y.D] \quad [A]}{\forall Y.D} \\
 D \\
 \hline
 A \rightarrow D
 \end{array}
 \quad
 \frac{
 \frac{
 \frac{[B \rightarrow \forall Y.D] \quad [B]}{\forall Y.D} \\
 D \\
 \hline
 B \rightarrow D
 \end{array}
 }{
 \frac{
 \frac{D}{\forall Y.D} \\
 (B \rightarrow \forall Y.D) \rightarrow \forall Y.D \\
 \hline
 (A \rightarrow \forall Y.D) \rightarrow ((B \rightarrow \forall Y.D) \rightarrow \forall Y.D)
 \end{array}
 }{D}$$

above the formula  $(A \rightarrow C) \rightarrow ((B \rightarrow C) \rightarrow C)$ . By induction hypothesis, the derivation reduces to

$$\frac{\begin{array}{c} \vdots \\ \vdots \\ A \quad [A \rightarrow \forall Y.D] \\ \hline \forall Y.D \\ D \\ \hline \forall Y.D \\ (B \rightarrow \forall Y.D) \rightarrow \forall Y.D \end{array}}{\frac{(A \rightarrow \forall Y.D) \rightarrow ((B \rightarrow \forall Y.D) \rightarrow \forall Y.D)}{(B \rightarrow \forall Y.D) \rightarrow \forall Y.D}} \quad \frac{\begin{array}{c} [A] \\ \vdots \\ \vdots \\ \forall Y.D \\ A \rightarrow \forall Y.D \end{array}}{\forall Y.D} \quad \frac{\begin{array}{c} [B] \\ \vdots \\ \vdots \\ \forall Y.D \\ B \rightarrow \forall Y.D \end{array}}{B \rightarrow \forall Y.D}$$

via  $\beta\eta$ -conversions. We only changed the portion of derivation above  $D$ . Applying

two arrow  $\beta$ -conversions we obtain  $\frac{\begin{array}{c} \vdots \\ \vdots \\ A \quad \frac{\forall Y.D}{A \rightarrow \forall Y.D} \\ \hline \frac{\forall Y.D}{D} \\ \hline \forall Y.D \end{array}}$  and with one more

arrow  $\beta$ -conversion we get  $\frac{\begin{array}{c} \vdots \\ \vdots \\ A \\ \hline \frac{\forall Y.D}{D} \\ \hline \forall Y.D \end{array}}$ . Finally, with one universal  $\eta$ -conversion the

derivation reduces to  $\frac{\begin{array}{c} \vdots \\ \vdots \\ A \\ \hline \forall Y.D \end{array}}$ . This concludes the proof. +

PROPOSITION 2. *The canonical translation of*

$$\frac{\begin{array}{c} \vdots \\ \vdots \\ A \\ \hline A \vee B \end{array} \quad \frac{\begin{array}{c} [A] \\ \vdots \\ \vdots \\ C \end{array} \quad \frac{\begin{array}{c} [B] \\ \vdots \\ \vdots \\ C \end{array}}{C}}{C}$$

*reduces to the canonical translation of*

$$\frac{\begin{array}{c} \vdots \\ \vdots \\ A \\ \hline C \end{array}}{C}$$

*using  $\beta\eta$ -conversions.*

PROOF. The canonical translation of the redex of the standard conversion for disjunction has exactly the form of the derivation in Lemma 4. The proof follows.  $\dashv$

The standard conversion associated with conjunction can be studied in a similar manner. Alternatively, we can rely on the observation that the connective of conjunction (as opposed to disjunction) is a “good” connective (see the enjoyable and interesting discussions in Chapter 10 of [6]). A straightforward reducibility clause can be defined for conjunction and the strong  $\beta\eta$ -normalization proof of  $\mathbf{F}_{\text{at}}$  generalizes easily to a system extended with the *primitive* symbol of conjunction.

THEOREM 2. *The intuitionistic natural deduction calculus of  $\wedge, \vee, \rightarrow, \perp$  with the standard ( $\beta$ ) conversions is strongly normalizable.*

PROOF. From Proposition 2 and the above discussion, we know that each  $\beta$ -reduction step of the full propositional calculus translates into at least one  $\beta\eta$ -reduction step of the system  $\mathbf{F}_{\text{at}}$ . Therefore, the result follows from the strong normalization of  $\mathbf{F}_{\text{at}}$  with respect to  $\beta\eta$ -conversions (see Corollary 1 above).  $\dashv$

**§5. Comments, questions, opinions.** The study of predicative variants of system  $\mathbf{F}$  is by no means a novel subject. As far as we are aware, there are two basic approaches to the subject. One is based on the restriction on the types (formulas) allowed in the language of  $\mathbf{F}$ . This is the approach of [2] and [1], where restrictions concerning parameters and nesting are in place in the language (the system studied in the former reference is predicatively reducible, and the system of the latter reference goes beyond predicativity, but subsumes the first system as a particular case). The other approach restricts the elimination rule of the universal second-order quantifier. For instance, John Mitchell describes in [9] a predicative polymorphic lambda calculus  $\lambda^{\rightarrow, \Pi}$ . In this calculus, impredicativism is eliminated by means of a division of the types into two universes: The first universe is constructed from base types and type variables using  $\rightarrow$ , and the second universe results from the first by closing it with respect to universal quantifications ranging over variables of the *first* universe. Several predicative extensions of  $\lambda^{\rightarrow, \Pi}$  can be found in the literature [10, 9, 8]. These strategies can ultimately be traced to ideas of Bertrand Russell and Alfred North Whitehead when they introduced *ramified* types in their *magnum opus* “Principia Mathematica” [14]. Our approach is the second, but with a twist. It is observed that a property of reducibility (here in a sense related to that of the axiom of reducibility of Russell and Whitehead, not to be confounded with the notion of reducibility in normalization proofs) comes for free in our setting via the phenomenon of instantiation overflow.

The measure of instantiation overflow that we discussed was restricted to universal second-order quantifications of the form

$$\begin{aligned} \perp &::= \forall X.X \\ A \wedge B &::= \forall X((A \rightarrow (B \rightarrow X)) \rightarrow X) \end{aligned}$$

$$A \vee B := \forall X((A \rightarrow X) \rightarrow ((B \rightarrow X) \rightarrow X))$$

The above three types correspond to the empty type, the product type and the sum type (respectively) in the terminology of Girard *et al.* [6]. We believe that it is an interesting question to characterize exactly which types enjoy the property of instantiation overflow.

By viewing the full intuitionistic calculus embedded in  $\mathbf{F}_{\text{at}}$ , we were able to find a rather natural proof of strong normalization for it. The proof is based on two algorithms: One underlies the embedding of the full intuitionistic propositional calculus into  $\mathbf{F}_{\text{at}}$  (relying heavily on instantiation overflow), and the other is related to the transformation of the  $\beta$ -conversions of the full intuitionistic propositional calculus into  $\beta\eta$ -reductions of  $\mathbf{F}_{\text{at}}$ . It would be nice to have a better understanding of these algorithms, but we did not pursue this line of research in the present paper. We also want to draw attention to the fact that our proof bestows a crucial role to  $\eta$ -conversions. In Section 3.2 of [6], Girard says that  $\eta$ -conversions “have never been given adequate status.” We cannot but help wondering if the role of  $\eta$ -conversions in our strong normalization proof could not indicate a way of providing a reasonable status for  $\eta$ -conversions. The concrete proof that we presented follows (as commented) Tait’s method of reducibility and, as such, is not formalizable in Peano arithmetic (let alone in primitive recursive arithmetic). This is incidental, we think. It would be nice to investigate whether, for instance, the proof technique used by Felix Joachimski and Ralph Matthes in [7] also applies to our system  $\mathbf{F}_{\text{at}}$  of atomic polymorphism.

We have been referring to [6] often. This is not only because we find this work of Girard *et al.* very well-written and clear, but also because it raises interesting issues and contains some rather stimulating comments. Girard defends that the elimination rules for  $\perp$  and  $\vee$  are not as natural and as well behaved as the other inference rules. It is even added that the presence of these elimination rules poses problems (e.g., the need for commuting conversions in order to have the subformula property in normal proofs) and gives rise to such “boring complications” that “one tends to think that natural deduction should be modified to correct such atrocities” (*ibidem* Section 10.5). System  $\mathbf{F}_{\text{at}}$  (which embeds intuitionistic calculus and, in particular,  $\perp$  and  $\vee$ ), on the other hand, has very well-behaved rules and, moreover, possesses a sensible notion of subformula. It is not difficult to see, by induction on the number of inferences in a normal proof (consult Subsection 10.3.1 of [6] for such kind of argument), that the following is true:

Let  $\Delta$  be a normal proof in  $\mathbf{F}_{\text{at}}$ . Then:

- (i) Every formula in  $\Delta$  is a subformula of the conclusion or of an undischarged hypothesis.
- (ii) If  $\Delta$  ends in an elimination rule, then there is a (so-called) principal branch  $(A_0, \dots, A_n)$  such that
  - $A_0$  is an undischarged hypothesis
  - $A_n$  is the conclusion

- $A_i$  is a principal premise of an elimination rule with conclusion  $A_{i+1}$ .

In particular,  $A_n$  is a subformula of  $A_0$ .

From the above, one can prove standard results in structural proof theory (e.g., see [3] for the disjunction property). We have suggested in [4] that  $\mathbf{F}_{\text{at}}$  provides a natural and appealing framework for full intuitionistic propositional logic. As Girard says in Section 10.5 of [6], “(...) the  $(\perp, \vee, \exists)$  fragment of the calculus is (not) etched on tablets of stone.” The exact determination of the proof-theoretical strength of  $\mathbf{F}_{\text{at}}$  may be relevant for the appreciation of our proposal. This seems to be an interesting problem.

In spite of the comments in the above paragraph, one can still pose the question whether it is possible to produce, within our framework, an argument for the strong normalization of the full propositional intuitionistic calculus *with* commuting conversions (or with  $\eta$ -conversions). In [4], it was proved that (the translation of) a *redex* and (the translation of) its *contractum*, with respect to a commuting conversion, are  $\beta$ -equivalent in  $\mathbf{F}_{\text{at}}$ . In other words, for commuting conversions, one can go from the *redex* to the *contractum* by means of  $\beta$ -conversions in *both* directions. The question at the beginning of this paragraph is, nevertheless, still not answered.

#### REFERENCES

- [1] K. AEHLIG, *Parameter-free polymorphic types*, *Annals of Pure and Applied Logic*, vol. 156 (2008), pp. 3–12.
- [2] T. ALTENKIRCH and T. COQUAND, *A finitary subsystem of the polymorphic lambda-calculus, Typed lambda calculi and applications (TLCA 2001)* (S. Abramsky, editor), Lecture Notes in Computer Science, vol. 2044, 2001, pp. 22–28.
- [3] F. FERREIRA, *Comments on predicative logic*, *Journal of Philosophical Logic*, vol. 35 (2006), pp. 1–8.
- [4] F. FERREIRA and G. FERREIRA, *Commuting conversions vs. the standard conversions of the “good” connectives*, *Studia Logica*, vol. 92 (2009), pp. 63–84.
- [5] J.-Y. GIRARD, *Une extension de l’interprétation de Gödel à l’analyse, et son application à l’élimination des coupures dans l’analyse et la théorie des types*, *Proceedings of the second scandinavian logic symposium* (J. E. Fenstad, editor), North Holland, Amsterdam, 1971, pp. 63–92.
- [6] J.-Y. GIRARD, Y. LAFONT, and P. TAYLOR, *Proofs and types*, Cambridge University Press, 1989.
- [7] F. JOACHIMSKI and R. MATTHES, *Short proofs of normalization for the simply-typed lambda-calculus, permutative conversions and Gödel’s T*, *Archive for Mathematical Logic*, vol. 42 (2003), pp. 59–87.
- [8] M. KRETZ, *On the treatment of predicative polymorphism in theories of explicit mathematics*, *Ph.D. thesis*, Universität Bern, 2002.
- [9] J. C. MITCHELL, *Type systems for programming languages*, *Handbook of theoretical computer science* (J. van Leeuwen, editor), vol. B, Elsevier, 1990, pp. 365–458.
- [10] J. C. MITCHELL and R. HARPER, *The essence of ML*, *Proceedings of 15th ACM symp. on principles of programming languages*, 1988, pp. 28–46.
- [11] D. PRAWITZ, *Ideas and results in proof theory*, *Proceedings of the second scandinavian logic symposium* (J. E. Fenstad, editor), North-Holland, 1971, pp. 235–307.



- [12] ———, *Natural deduction*, Almqvist & Wiksell, Stockholm, 1965. Reprinted, with a new preface, in Dover Publications, 2006.
- [13] J. C. REYNOLDS, *Towards a theory of type structure, Colloque sur la programmation, LNCS 19* (B. Robinet, editor), Springer-Verlag, 1974.
- [14] B. RUSSELL and A. N. WHITEHEAD, *Principia mathematica*, Cambridge University Press (2nd edition), 1927.
- [15] T. SANDQVIST, *A note on definability of logical operators in second-order logic*, Unpublished manuscript, 2008.
- [16] W. TAIT, *Intentional interpretations of functionals of finite type I*, *The Journal of Symbolic Logic*, vol. 32 (1967), pp. 198–212.
- [17] A. S. TROELSTRA and H. SCHWICHTENBERG, *Basic proof theory*, Cambridge University Press, Cambridge, 1996.

FACULDADE DE CIÊNCIAS DA UNIVERSIDADE DE LISBOA  
DEPARTAMENTO DE MATEMÁTICA  
CAMPO GRANDE, ED. C6, 1749-016 LISBOA, PORTUGAL  
*E-mail*: fferreira@fc.ul.pt

UNIVERSIDADE LUSÓFONA DE HUMANIDADES E TECNOLOGIAS  
DEPARTAMENTO DE MATEMÁTICA  
AV. DO CAMPO GRANDE, 376, 1749-024 LISBOA, PORTUGAL  
*E-mail*: gildafer@cii.fc.ul.pt