J. Functional Programming **6** (3): 519–525, May 1996 © 1996 Cambridge University Press

Proof-irrelevance out of excluded-middle and choice in the calculus of constructions

FRANCO BARBANERA AND STEFANO BERARDI

Dipartimento di Informatica, Universita' di Torino, Corso Svizzera 185, 10149 Torino, Italy
(e-mail: {barba,stefano} @di.unito.it*)*

Abstract

We present a short and direct syntactic proof of the fact that adding the axiom of choice and the principle of excluded-middle to Coquand–Huet's Calculus of Constructions gives proof-irrelevance.

Capsule Review

Systems of type theory are used to formalise proofs. For such formalisations it is always important to know whether they are faithful, i.e. if there exists a formal proof, is there also an proof in intuitive mathematics? The present paper does not directly adress this issue but is relevant: if the axiom of choice and the excluded middle are added to certain type systems, then all proofs become provably equal. This is in the spirit of classical mathematics, in which one cares more about the truth of a statement than about its provability. The property does not only hold for the calculus of constructions but also for simpler systems in the λ-cube. The proof does not carry over to the logical-cube.

1 Introduction

The present paper aims to be a contribution to the analysis of the intensional aspects of Coquand–Huet's Calculus of Constructions (CC). We will show that adding the classical principle of excluded-middle ($\mathscr{EM}$) and the axiom of choice ($\mathscr{AC}$) to CC yields the so called proof-irrelevance, i.e. any small class (element of Prop) has at most one element for Leibniz equality or, equivalently, the only model for the theory turns out to be necessarily the one where Prop is interpreted as a two elements set and each type is either empty or it has only one element.

Proofs of similar results were previously devised by Coquand and Pottinger. Coquand (1990) showed that CC with the excluded-middle and a derivation rule for a strong version of disjoint sum yields proof-irrelevance. To do so he showed that, by assuming true to be not equal to false (with respect to Leibniz equality), it is possible to define an interpretation of Girard's logically inconsistent system, $\mathscr{U}$, into his extended CC. Thus, by the inconsistency of $\mathscr{U}$, it follows that true is equal to false, a fact that implies proof-irrelevance.

https://doi.org/10.1017/S0956796800001829 Published online by Cambridge University Press

A similar argument was also used by Pottinger (1989) (see also Geuvers (1993, pp. 158–159)) to show that proof-irrelevance can also be obtained by extending CC with the excluded-middle and the so-called axiom of definite descriptions, which can be seen as a weaker form of the axiom of choice.

The above-mentioned proofs then heavily rely on the properties of Girard's system $\mathscr{U}$, while the alternative proof we give here is direct and self-contained in nature. It will consist in constructing a small class and an embedding-projection pair showing that the set of its parts is a retract of it. The existence of such a pair will then be proved to imply a formalisation of Russell's paradox, and hence the identification, with respect to Leibniz equality, of true and false. (A related proof technique is also exploited by Hurkens (1995) to define a paradoxical term for a restriction of $\mathscr{U}$.)

Our proof will make essential use of the ambiguity, typical of CC, between small classes and propositions. This ambiguity will allow us to define, in section 3.1, a small class as a product indexed over all small classes. So, a similar proof could not go through in systems like (higher-order) predicate logic or HA_ω, even though they are close to CC in many aspects.

We shall use a strong version of the axiom of choice to enhance the simplicity of the proof, even if the result holds also for weak versions of it.

A complete formal description of our proof can easily be given in few lines of code in any current implementation of CC. In particular, the formalisation in the LEGO language of a variant of it can be found in Berardi (1994).

A close examination of our proof shows that it works not only for the whole CC, but it can also be carried over in a fragment of CC, i.e. its subsystem $\lambda P2$ (see Barendregt (1992) and Geuvers (1994) for a definition of $\lambda P2$ and its relationship to logics).

In what follows we assume the reader to be well acquainted with CC and its properties. We refer otherwise to Barendregt (1992), Coquand and Huet (1988) and Geuvers (1993).

2 Preliminaries

We recall that it is possible to define all the logical operators inside CC, as well as their introduction and elimination rules in natural deduction. For instance, we can define $\exists x{:}A.P =_{Def} \Pi X{:}\mathsf{Prop}.\big((\Pi x{:}A.P \to X) \to X\big)$, where $B \to C$ is short, as usual, for $\Pi y{:}B.C$ with $y \notin FV(C)$. In the following, the terms in CC that represent binary logical operations ('$\wedge$','$\vee$',...) will be used, for the sake of readability, in infix notation.

The Leibniz equality can, as usual, be defined as follows:

$$\mathsf{Eq} =_{Def} \lambda A{:}\mathsf{Prop}.\lambda x\, y{:}A.\Pi P{:}(A \to \mathsf{Prop}).(Py) \to (Px).$$

In the rest of the paper we shall write simply '$a =_A b$' for (Eq A a b), where A is the type of a and b.

One definition more we recall is that of the type of Booleans and its elements:

$$\mathsf{Bool} =_{Def} \Pi A{:}\mathsf{Prop}.A \to A \to A,$$

https://doi.org/10.1017/S0956796800001829 Published online by Cambridge University Press

$$\mathsf{true} =_{Def} \lambda A{:}\mathsf{Prop}.\lambda x\, y{:}A.x \quad , \quad \mathsf{false} =_{Def} \lambda A{:}\mathsf{Prop}.\lambda x\, y{:}A.y.$$

To have an inhabitant of the type $\mathsf{false} =_{\mathsf{Bool}} \mathsf{true}$ implies, as said above, proof-irrelevance, that is $a =_A b$ for any small type A and $a, b : A$. In fact, for any A and $a, b : A$, it is possible to define $\mathsf{f} : \mathsf{Bool} \to A$ such that $(\mathsf{f}\ \mathsf{true})$ is convertible to a and $(\mathsf{f}\ \mathsf{false})$ is convertible to b.

Giving explicitely all the CC-terms that form our proof could prevent the reader to single out the main ideas underlying it. So we shall often describe how to build such CC-terms, without actually doing it. To make our argument still more readable we shall often use the abbreviation 'derive A' for 'derive a term of type A'.

2.0.1 Formalising $\mathscr{EM}$ and $\mathscr{AC}$ into CC

Let us see how to formalise in CC the law of the excluded-middle and the axiom of choice. For the excluded-middle we can simply introduce a constant $\mathscr{EM}$ which, taken an element A of Prop returns an element of $(A \vee \neg A)$, i.e.

$$\mathscr{EM} : \Pi A{:}\mathsf{Prop}.(A \vee \neg A).$$

It is worth recalling that introducing a constant in CC corresponds to working in a context with a variable of suitable type.

In CC, sets are necessarily introduced by comprehension, so the axiom of choice can be seen as the possibility of getting, out of a proof of $\exists x{:}A.P(x)$, an element a of A and a proof of $P(a)$. In particular, we can introduce two constants that, given $A : \mathsf{Prop}$, a predicate P on A and a term (proof) $\mathsf{h} : \exists x{:}A.(P\ x)$ return, respectively, a witness a of type A and a proof of $(P\ a)$, i.e. of the fact that a is indeed a witness of $\exists x{:}A.(Px)$.

We therefore add the following two constants to CC:

$$\mathscr{AC}_{\mathscr{F}} : \Pi A{:}\mathsf{Prop}.\Pi P{:}(A \to \mathsf{Prop}).\big((\exists x{:}A.(Px)) \to A\big)$$

$$\mathscr{AC}_{\mathscr{A}} : \Pi A{:}\mathsf{Prop}.\Pi P{:}(A \to \mathsf{Prop}).\Pi h{:}\exists x{:}A.(Px).(P(\mathscr{AC}_{\mathscr{F}}\ h))$$

where $(\mathscr{AC}_{\mathscr{F}}\ h)$ is short for $(\mathscr{AC}_{\mathscr{F}}\ A\ P\ h)$.

Note that these terms are uniform on A and P. So, our axiom of choice is stronger than the choice axiom of set theory. Moreover, if one defines $\exists!x{:}A.(Px)$ as $(\exists x{:}A.(Px)) \wedge (\Pi x\, y{:}A.(Px) \to (Py) \to (x =_A y))$, and replaces $\exists$ by $\exists!$, then $\mathscr{AC}_{\mathscr{F}}$ and $\mathscr{AC}_{\mathscr{A}}$ become weaker and correspond, respectively, to the axiom ι and to the axiom of definite-descriptions described in Pottinger (1989) (see also Geuvers (1993, pp. 158–159)).

2.0.2 Reasoning by cases in CC+$\mathscr{EM}$+$\mathscr{AC}$

Since it will be necessary, in our proof, to reason by cases, we internalise in CC +$\mathscr{EM}$+$\mathscr{AC}$ such sort of reasoning by building an operator IfThenElse that, given an element $A : \mathsf{Prop}$, considered as condition, and two elements c_1 and c_2 of a type $C : \mathsf{Prop}$, 'returns' (is equal to) c_1 in case A is true (inhabited), and c_2 otherwise. It is easy to realise that in the definition of such an operator, the use of the axiom $\mathscr{EM}$ plays an essential rôle.

https://doi.org/10.1017/S0956796800001829 Published online by Cambridge University Press

Given A : Prop and $c_1, c_2 : C$, we may build, by means of the formalisation of a simple standard argument in classical logic, a term (proof) p of type $\exists x{:}C.\big(((x =_C c_1) \wedge A) \vee ((x =_C c_2) \wedge (\neg A))\big)$. We can then 'extract' from p, by means of $\mathscr{AC}_{\mathscr{F}}$, a term $(\mathscr{AC}_{\mathscr{F}}\ \mathsf{p}) : C$ and, using $\mathscr{AC}_{\mathscr{A}}$, prove it equal to c_1 or c_2 in case A is true or false (i.e. inhabited or not), respectively. In particular, it is possible to prove $A \to ((\mathscr{AC}_{\mathscr{F}}\ \mathsf{p}) =_C c_1)$ and $\neg A \to ((\mathscr{AC}_{\mathscr{F}}\ \mathsf{p}) =_C c_2)$.
Our term IfThenElse will then be

$$\mathsf{IfThenElse} =_{Def} \lambda A\, C{:}\mathsf{Prop}.\lambda c_1\, c_2{:}C.(\mathscr{AC}_{\mathscr{F}}\ \mathsf{p}).$$

In the following we shall write (IfThenElse $A\ c_1\ c_2$), instead of (IfThenElse $A\ C\ c_1\ c_2$), to enhance readability.

3 CC+$\mathscr{EM}$+$\mathscr{AC}$ $\vdash$ *Proof-Irrelevance*

In this section we present the proof of our main result. It will consist of two main steps:

- The construction of a small class U : Prop and of an embedding-projection pair showing that $\mathscr{P}$(U) (the 'powerset of U') is a retract of U.
- The derivation of Russell's paradox from the existence of the above embedding-projection pair between $\mathscr{P}$(U) and U.

3.1 U *and the embedding-projection* $\mathscr{P}(\mathsf{U}) \lhd \mathsf{U}$

Let us begin by defining the small class (proposition) U as the infinite product of all the powersets of small classes.

$$\mathsf{U} =_{Def} \Pi X{:}\mathsf{Prop}.\mathscr{P}(X) : \mathsf{Prop},$$

where $\mathscr{P}(X)$ is short for the proposition $(X \to \mathsf{Bool}) : \mathsf{Prop}$, denoting the set of 'subsets' of X.

It is clear that $\mathscr{P}$(U), being an element of Prop, is a 'component' of U itself and hence it is possible to define an embedding $\mathscr{P}(\mathsf{U}) \hookrightarrow \mathsf{U}$. In fact, given an element f in $\mathscr{P}$(U), we can map it into the infinite sequence having f at 'position' U and an arbitrary value of $\mathscr{P}(X)$ at any other 'position' X.

It is also easy to see that U can be trivially mapped onto $\mathscr{P}$(U): given an infinite sequence contained in U one can simply 'select' its component at 'position' U. So, it is intuitively possible to show that $\mathscr{P}(\mathsf{U}) \lhd \mathsf{U}$, i.e. that $\mathscr{P}$(U) is a retract of U, by means of the embedding-projection pair informally described above.

Let us actually build now the terms in CC+$\mathscr{EM}$+$\mathscr{AC}$ that formalise it.

The projection is trivial to describe formally.

$$\mathsf{proj}_{\mathsf{U}} =_{Def} \lambda u{:}\mathsf{U}.(u\mathsf{U}) : \mathsf{U} \to \mathscr{P}(\mathsf{U}).$$

The formalisation of the embedding, instead, is not so immediate. Let $f : \mathscr{P}(\mathsf{U})$. Following the informal description of the embedding, we can map f to $\lambda X{:}\mathsf{Prop}.\mathsf{q}_X$:

https://doi.org/10.1017/S0956796800001829 Published online by Cambridge University Press

U, where the term q_X is such that

$$\mathsf{q}_X = \begin{cases} f & \text{if } X \text{ is } \mathsf{U} \\ \mathsf{anything}_X & \text{otherwise.} \end{cases}$$

$\mathsf{anything}_X$ can be any element of $\mathscr{P}(X)$ in the product $\mathsf{U}(\equiv \Pi X{:}\mathsf{Prop}.\mathscr{P}(X))$. We could choose it to be, for instance, $\lambda x{:}X.\mathsf{false}$, i.e. the empty subset of X, henceforth denoted by $\emptyset_X : \mathscr{P}(X)$.

Then our goal is now to get a formal description of the term q_X.

Note that the naive definition of q_X as $(\mathsf{IfThenElse}\,(X \cong \mathsf{U})\ f\ \emptyset_X)$, where '$\cong$' is the CC-predicate stating that two small classes are isomorphic, cannot work because of typing problems. In fact f and $\emptyset_X$, which should have the very same type in $(\mathsf{IfThenElse}\,(X \cong \mathsf{U})\ f\ \emptyset_X)$, have type $\mathscr{P}(\mathsf{U})$ and $\mathscr{P}(X)$, respectively. To get q_X we can, instead, proceed as follows. Let us consider the CC-predicate stating that two small classes are one the retract of the other.

$$\text{`}\lhd\text{'} =_{Def} \lambda A\,B{:}\mathsf{Prop}.\exists g{:}A \to B.\exists h{:}B \to A.(\mathsf{e\text{-}p\text{-}pair}\ g\ h),$$

where e-p-pair, the predicate stating that two functions form an embedding-projection pair, is defined as follows.

$$\mathsf{e\text{-}p\text{-}pair} =_{Def} \lambda g{:}A \to B.\lambda h{:}B \to A.\big((h \circ g) =_{A \to A} (\mathsf{Id}\ A)\big).$$

Id is the obvious CC-term representing the polymorphic identity and '$\circ$', used in infix notation, is the CC-term denoting function composition. Also '$\lhd$' will be used in infix notation.

From the definition of '$\lhd$' it is straightforward to get a term, say t_X, of type

$$(\mathscr{P}(X) \lhd \mathscr{P}(\mathsf{U})) \to \big(\exists g{:}\mathscr{P}(X) \to \mathscr{P}(\mathsf{U}).\exists h{:}\mathscr{P}(\mathsf{U}) \to \mathscr{P}(X).(\mathsf{e\text{-}p\text{-}pair}\ g\ h)\big). \qquad (1)$$

In classical logic, unlike in the intuitionistic case, any formula of the form $(Q \to \exists x.S)$, with $x \notin FV(Q)$, is equivalent to $\exists x.(Q \to S)$ in case the domain of x is non-empty. Now, since in CC+$\mathscr{EM}$+$\mathscr{AC}$ we can formalise classical reasoning, and the domains $\mathscr{P}(X) \to \mathscr{P}(\mathsf{U})$ and $\mathscr{P}(\mathsf{U}) \to \mathscr{P}(X)$ are both non-empty (they contain, for instance, the constant functions $\lambda x{:}\mathscr{P}(X).\emptyset_{\mathsf{U}}$ and $\lambda x{:}\mathscr{P}(U).\emptyset_X$, respectively), it is easy to get a term, say t'_X, of type

$$\exists g{:}\mathscr{P}(X) \to \mathscr{P}(\mathsf{U}).\exists h{:}\mathscr{P}(\mathsf{U}) \to \mathscr{P}(X).\big((\mathscr{P}(X) \lhd \mathscr{P}(\mathsf{U})) \to (\mathsf{e\text{-}p\text{-}pair}\ g\ h)\big). \qquad (2)$$

Now, by using $\mathscr{AC}_{\mathscr{F}}$ and $\mathscr{AC}_{\mathscr{A}}$ on t'_X, it is possible to get a pair of terms $\langle \phi_X, \psi_X \rangle$, and prove that they are an embedding-projection pair in case $\mathscr{P}(X) \lhd \mathscr{P}(\mathsf{U})$ (a condition trivially satisfied when X is U). In particular, $(\psi_X(\phi_{\mathsf{U}}\ f))$ is the term q_X we were looking for, i.e. a term of type $\mathscr{P}(X)$ equal to f in case X is U (if it returns $\emptyset_X$ or something else otherwise, is not essential for our argument).

Then we have that $\mathscr{P}(\mathsf{U}) \lhd \mathsf{U}$ by means of the embedding-projection pair $\langle \mathsf{inj}_{\mathsf{U}}, \mathsf{proj}_{\mathsf{U}} \rangle$, where

$$\mathsf{inj}_{\mathsf{U}} =_{Def} \lambda f{:}\mathscr{P}(\mathsf{U}).\lambda X{:}\mathsf{Prop}.(\psi_X(\phi_{\mathsf{U}}\ f)) : \mathscr{P}(\mathsf{U}) \to \mathsf{U}$$

$$\mathsf{proj}_{\mathsf{U}} =_{Def} \lambda u{:}\mathsf{U}.(u\mathsf{U}) : \mathsf{U} \to \mathscr{P}(\mathsf{U}).$$

https://doi.org/10.1017/S0956796800001829 Published online by Cambridge University Press

In fact, since ϕ_X and ψ_X come out of t'_X we can derive

$$\lambda f{:}\mathscr{P}(\mathsf{U}).(\psi_{\mathsf{U}}\ (\phi_{\mathsf{U}}\ f)) =_{\mathscr{P}(\mathsf{U})\to\mathscr{P}(\mathsf{U})}\ (\mathsf{Id}\ \mathscr{P}(\mathsf{U})).$$

From this, using the fact that

$$(\mathsf{proj}_{\mathsf{U}} \circ \mathsf{inj}_{\mathsf{U}}) =_{\beta} \lambda f{:}\mathscr{P}(\mathsf{U}).(\psi_{\mathsf{U}}\ (\phi_{\mathsf{U}}\ f)) =_{\beta} (\psi_{\mathsf{U}} \circ \phi_{\mathsf{U}}),$$

it is then possible to derive

$$(\mathsf{proj}_{\mathsf{U}} \circ \mathsf{inj}_{\mathsf{U}}) =_{\mathscr{P}(\mathsf{U})\to\mathscr{P}(\mathsf{U})}\ (\mathsf{Id}\ \mathscr{P}(\mathsf{U})).$$

3.2 $\mathscr{P}(\mathsf{U}) \lhd \mathsf{U}$ *implies* ***Russell's paradox***

Using U and the fact that $\mathscr{P}(\mathsf{U})$ is a retract of it, it is possible to rephrase in our setting the well known argument of Russell's paradox.

The universe we consider is U. The existence of a projection from U to $\mathscr{P}(\mathsf{U})$ enables us to consider the elements of U also as sets (an element $u : \mathsf{U}$ corresponds to the set $(\mathsf{proj}_{\mathsf{U}}\ u) : \mathscr{P}(\mathsf{U})$). So, we can express the fact that $u : \mathsf{U}$, seen as element, 'belongs' to $v : \mathsf{U}$, seen as set, by means of the following CC-predicate

$$\mathsf{belongs} =_{Def} \lambda u\, v{:}\mathsf{U}.\big(((\mathsf{proj}_U\ v)\ u) =_{\mathsf{Bool}} \mathsf{true}\big) : \mathsf{U} \to \mathsf{U} \to \mathsf{Prop}.$$

We will denote $(\mathsf{belongs}\ u\ v)$ by $(u \in v)$.

We can now carry on Russell's argument by defining the collection of the elements of our universe that do not belong to themselves.

$$\mathsf{RussellClass} =_{Def} \lambda u{:}\mathsf{U}.\big(\mathsf{IfThenElse}\ \neg(u \in u)\ \mathsf{true}\ \mathsf{false}\big) : \mathscr{P}(\mathsf{U}).$$

The element of U corresponding to the element of $\mathscr{P}(\mathsf{U})$ just defined is then

$$\mathsf{r} =_{Def} (\mathsf{inj}_{\mathsf{U}}\ \mathsf{RussellClass}) : \mathsf{U}.$$

Such an object, as in Russell's argument, is paradoxical. In fact it is possible to derive, as shown below, $(\mathsf{false} =_{\mathsf{Bool}} \mathsf{true})$ both from the assumption $(\mathsf{r} \in \mathsf{r})$ and from the assumption $\neg(\mathsf{r} \in \mathsf{r})$. Moreover, by $\mathscr{EM}$, we can derive $\big((\mathsf{r} \in \mathsf{r}) \vee \neg(\mathsf{r} \in \mathsf{r})\big)$, and hence, by means of the term formalising in CC the rule of elimination of disjunction, it is possible to obtain a closed term of type $(\mathsf{false} =_{\mathsf{Bool}} \mathsf{true})$. Such a term, as said before, gives proof-irrelevance.

Let us see how to get $(\mathsf{false} =_{\mathsf{Bool}} \mathsf{true})$ both from $(\mathsf{r} \in \mathsf{r})$ and from $\neg(\mathsf{r} \in \mathsf{r})$.

Let us assume $(\mathsf{r} \in \mathsf{r})$. By expanding the definitions of r and $\in$, we get

$$\big((\mathsf{proj}_U\ (\mathsf{inj}_{\mathsf{U}}\ \mathsf{RussellClass}))\ \mathsf{r}\big) =_{\mathsf{Bool}} \mathsf{true}.$$

Now, since $\langle \mathsf{inj}_{\mathsf{U}}, \mathsf{proj}_{\mathsf{U}} \rangle$ is an embedding-projection pair, i.e. $(\mathsf{proj}_{\mathsf{U}} \circ \mathsf{inj}_{\mathsf{U}}) = {}_{\mathscr{P}(\mathsf{U})\to\mathscr{P}(\mathsf{U})}(\mathsf{Id}\ \mathscr{P}(\mathsf{U}))$, we get $\big((\mathsf{RussellClass}\ \mathsf{r}) =_{\mathsf{Bool}} \mathsf{true}\big)$. This, by definition, is

$$((\mathsf{IfThenElse}\ \neg(\mathsf{r} \in \mathsf{r})\ \mathsf{true}\ \mathsf{false}) =_{\mathsf{Bool}} \mathsf{true})$$

and hence, by our assumption and the properties of IfThenElse, we can derive $(\mathsf{false} =_{\mathsf{Bool}} \mathsf{true})$.

Now let us assume instead $\neg(\mathsf{r} \in \mathsf{r})$. Following the same argument of the previous

https://doi.org/10.1017/S0956796800001829 Published online by Cambridge University Press

case it is possible, this time, to get $\neg(\mathsf{true} =_{\mathsf{Bool}} \mathsf{true})$ and $\bot$ from it. By *ex-falso-quodlibet*, also in this case we derive $(\mathsf{false} =_{\mathsf{Bool}} \mathsf{true})$.

It is worth noticing that indeed, for the Russel's paradox, we do not really need $\mathsf{inj}_{\mathsf{U}}$, but only the fact that $\mathsf{proj}_{\mathsf{U}}$ is surjective. To prove this fact one could proceed as follows. Consider the function $\psi_X : \mathscr{P}(\mathsf{U}) \to \mathscr{P}(X)$ as defined in the previous subsection. ψ_X is surjective whenever $\mathscr{P}(X)$ is a retract of $\mathscr{P}(U)$. Define now the following function.

$$F =_{Def} \lambda f{:}\mathscr{P}(\mathsf{U}).\lambda X{:}\mathsf{Prop}.(\psi_X\, f) : \mathscr{P}(\mathsf{U}) \to \mathsf{U}$$

By the fact that $(\mathsf{proj}_{\mathsf{U}} \circ F)$ is equal to ψ_{U} it follows that $\mathsf{proj}_{\mathsf{U}}$ is surjective as well.

It is also interesting to notice that in our argument we have derived proof-irrelevance by the possibility of deriving $(\mathsf{false} =_{\mathsf{Bool}} \mathsf{true})$, even if the definitions of Bool, true and false are not really necessary. In fact, it would be possible to work in a context containing the variables $A{:}\mathsf{Prop}, x{:}A, y{:}A$ and to replace in our argument A, x and y for Bool, true and false, respectively. We would then derive $x =_A y$, and hence proof-irrelevance.

Acknowledgements

We wish to thank Mariangiola Dezani and Mario Coppo for their support and gentle guidance, and an anonymous referee for helping us in improving section 3.1 and for the remarks at the end of section 3.2.

The first author wishes also to thank Eleonora and Pino Lucci.

References

Barendregt, H. P. (1992) Typed Lambda Calculi. In Abramski et al., editors, *Handbook of Logic in Computer Science.*, Oxford University Press.

Berardi, S. (1994) A formalization in LEGO of a proof of $\mathsf{CC}+\mathscr{EM}+\mathscr{AC} \vdash$ *Proof-Irrelevance.* Technical Report, Department of Computer Science University of Torino.

Coquand, T. (1990) Metamathematical investigation of a calculus of construction. In P. Odifreddi, editor, *Logic and Computer Science.* Academic Press.

Coquand, T. and Huet, G. (1988) The Calculus of Constructions. *Information and Computation* **76**.

Geuvers, H. (1993) Logics and Type Systems. PhD Thesis, University of Nijmegen.

Geuvers, H. (1994) Conservativity between logics and typed λ-calculi. *Proc. BRA Workshop on Types for Proofs and Programs.*

Hurkens, A. J. C. (1995) A simplification of Girard's paradox. In M. Dezani-Ciancaglini and G. Plotkin editors, *Proc. 2nd International Conference on Typed Lambda Calculi and Applications (TLCA'95): Lecture Notes in Computer Science 902.* Springer-Verlag.

Pottinger, G. (1989) Definite descriptions and excluded middle in the calculus of constructions. *TYPES network*, November.

https://doi.org/10.1017/S0956796800001829 Published online by Cambridge University Press