Continuation-passing Style Models Complete for Intuitionistic Logic

Danko Ilik

Continuation-passing Style Models Complete for Intuitionistic Logic

Danko Ilik

University "Goce Delčev" – Štip
Address: Faculty of Informatics, PO Box 201, Štip, Macedonia
E-mail: dankoilik@gmail.com

Abstract

A class of models is presented, in the form of continuation monads polymorphic for first-order individuals, that is sound and complete for minimal intuitionistic predicate logic. The proofs of soundness and completeness are constructive and the computational content of their composition is, in particular, a β-normalisation-by-evaluation program for simply typed lambda calculus with sum types. Although the inspiration comes from Danvy's type-directed partial evaluator for the same lambda calculus, the there essential use of delimited control operators (i.e. computational effects) is avoided. The role of polymorphism is crucial – dropping it allows one to obtain a notion of model complete for classical predicate logic. The connection between ours and Kripke models is made through a strengthening of the Double-negation Shift schema.

Key words: intuitionistic logic, completeness, Kripke models, Double-negation Shift, normalization by evaluation
2000 MSC: 03B20, 03B35, 03B40, 68N18, 03F55, 03F50, 03B55

1. Introduction

Although Kripke models are standard semantics for intuitionistic logic, there is as yet no (simple) constructive proof of their completeness when one considers all the logical connectives. While Kripke's original proof [20] was classical, Veldman gave an intuitionistic one [26] by using Brouwer's Fan Theorem to handle disjunction and the existential quantifier. To see what the computational content behind Veldman's proof is, one might consider a realisability interpretation of the Fan Theorem (for example [3]), but, all known realisers being defined by general recursion, due to the absence of an elementary proof of their termination, it is not clear whether one can think of the program using them as a constructive proof or not.

On the other hand, a connection between normalisation-by-evaluation (NBE) [4] for simply typed lambda calculus, $\lambda^{\to}$, and completeness for Kripke models for the fragment $\{\wedge, \Rightarrow, \forall\}$ has been made [6, 15]. We review this connection in Section 2. There we also look at Danvy's extension [8] of NBE from $\lambda^{\to}$ to $\lambda^{\to\vee}$, simply typed lambda calculus with sum types. Even though Danvy's algorithm is simple and elegant, he uses the full power of delimited control operators which do not yet have a

typing system that permits to understand them logically. We deal with that problem in Section 3, by modifying the notion of Kripke model so that we can give a proof of completeness for full intuitionistic logic in continuation-passing style, that is, without relying on having delimited control operators in our meta-language. In Section 4, we extract the algorithm behind the given completeness proof, a β-NBE algorithm for $\lambda^{\to\vee}$. In Section 5, we stress the importance of our models being dependently typed, by comparing them to similar models that are complete for classical logic [18]. We there also relate our and Kripke models by showing that the two are equivalent in presence of a strengthening of the Double-negation Shift schema [24, 25]. We conclude with Section 6 by mentioning related work.

The proofs of Section 3 have been formalised in the Coq proof assistant in [16], which also represents an implementation of the NBE algorithm.

2. Normalisation-by-Evaluation as Completeness

In [4], Berger and Schwichtenberg presented a proof of normalisation of $\lambda^{\to}$ which does not involve reasoning about the associated reduction relation. Instead, they interpreted λ-terms in a domain, or ambient meta-language, using an evaluation function,

$$[\![-]\!] : \Lambda \to D,$$

and then they defined an inverse to this function, which from the denotation in D directly extracts a term in $\beta\eta$-long normal form. The inverse function $\downarrow$, called *reification*, is defined by recursion on the type τ of the term, at the same time with an auxiliary function $\uparrow$, called *reflection*:

$$\begin{aligned}
&\downarrow^{\tau} : D \to \Lambda\text{-nf} && \\
&\downarrow^{\tau} := a \mapsto a && \tau\text{-atomic} \\
&\downarrow^{\tau\to\sigma} := S \mapsto \lambda a.\, \downarrow^{\sigma} (S \cdot \uparrow^{\tau} a) && a\text{-fresh}
\end{aligned}$$

$$\begin{aligned}
&\uparrow^{\tau} : \Lambda\text{-ne} \to D && \\
&\uparrow^{\tau} := a \mapsto a && \tau\text{-atomic} \\
&\uparrow^{\tau\to\sigma} := e \mapsto S \mapsto \uparrow^{\sigma} e(\downarrow^{\tau} S) &&
\end{aligned}$$

Here, S ranges over members of D, and we used $\mapsto$ and $\cdot$ for abstraction and application at the meta-level. The subclasses of normal and neutral λ-terms are given by the following inductive definition.

$$\begin{aligned}
&\Lambda\text{-nf} \ni r := \lambda a^{\tau}.r^{\sigma} \mid e^{\tau} && \lambda\text{-terms in normal form} \\
&\Lambda\text{-ne} \ni e := a^{\tau} \mid e^{\tau\to\sigma} r^{\tau} && \text{neutral } \lambda\text{-terms}
\end{aligned}$$

It was a subsequent realisation of Catarina Coquand [6], that the evaluation algorithm $[\![\cdot]\!]$ is also the one underlying the Soundness Theorem for minimal intuitionistic logic (with $\Rightarrow$ as the sole logical connective) with respect to Kripke models, and that the reification algorithm $\downarrow$ is also the one underlying the corresponding Completeness Theorem.

Definition 2.1. A *Kripke model* is given by a preorder $(K, \leq)$ of *possible worlds*, a binary relation of *forcing* $(-) \Vdash (-)$ between worlds and atomic formulae, and a *family of domains of quantification* $D(-)$, such that,

$$\begin{aligned} &\text{for all } w' \geq w, w \Vdash X \to w' \Vdash X, \text{ and} \\ &\text{for all } w' \geq w, D(w) \subseteq D(w'). \end{aligned}$$

The relation of forcing is then extended from atomic to composite formulae by the clauses:

$$\begin{aligned} w \Vdash A \wedge B &:= w \Vdash A \text{ and } w \Vdash B \\ w \Vdash A \vee B &:= w \Vdash A \text{ or } w \Vdash B \\ w \Vdash A \Rightarrow B &:= \text{for all } w' \geq w, w' \Vdash A \Rightarrow w' \Vdash B \\ w \Vdash \forall x.A(x) &:= \text{for all } w' \geq w \text{ and } t \in D(w'), w' \Vdash A(t) \\ w \Vdash \exists x.A(x) &:= \text{for some } t \in D(w), w \Vdash A(t) \\ w \Vdash \bot &:= \text{false} \\ w \Vdash \top &:= \text{true} \end{aligned}$$

More precisely, the following well-known statements hold and their proofs have been machine-checked [7, 15] for the logic fragment generated by the connectives $\{\Rightarrow, \wedge, \forall\}$.

Theorem 2.2 (Soundness)**.** *If* $\Gamma \vdash p : A$ *then, in any Kripke model, for any world* w*, if* $w \Vdash \Gamma$ *then* $w \Vdash A$*.*

Proof. By a simple induction on the length of the derivation. □

Theorem 2.3 (Strong Completeness by Substitution)**.** *There is a model* $\mathcal{U}$ *(the "universal model") such that, given a world* w *of* $\mathcal{U}$*, if* $w \Vdash A$*, then there exists a term* p *and a derivation in normal form* $w \vdash p : A$*.*

Proof. The universal model $\mathcal{U}$ is built by setting:

- K to be the set of contexts Γ;
- "$\leq$" to be the subset relation of contexts;
- "$\Gamma \Vdash X$" to be the set of derivations in normal form $\Gamma \vdash^{\mathrm{nf}} X$, for X an atomic formula.

One then proves simultaneously, by induction on the complexity of A, that the two functions defined above, reify ($\downarrow$) and reflect ($\uparrow$), are correct, that is, that $\downarrow$ maps a member of $\Gamma \Vdash A$ to a normal proof term (derivation) $\Gamma \vdash p : A$, and that $\uparrow$ maps a neutral term (derivation) $\Gamma \vdash e : A$ to a member of $\Gamma \Vdash A$. □ □

Corollary 2.4 (Completeness (usual formulation))**.** *If in any Kripke model, at any world* w*,* $w \Vdash \Gamma$ *implies* $w \Vdash A$*, then there exists a term* p *and a derivation* $\Gamma \vdash p : A$*.*

Proof. If $w \Vdash \Gamma \to w \Vdash A$ in any Kripke model, then also $w \Vdash \Gamma \to w \Vdash A$ in the model $\mathcal{U}$ above. Since from the $\uparrow$-part of Theorem 2.3 we have that $\Gamma \Vdash \Gamma$, then from the $\downarrow$-part of the same theorem there exists a term p such that $\Gamma \vdash p : A$. □ □

If one wants to extend this technique for proving completeness for Kripke models to the rest of the intuitionistic connectives, $\bot$, $\vee$ and $\exists$, the following meta-mathematical problems appear, which have been investigated in the middle of the last century. At that time, Kreisel, based on observations of Gödel, showed (Theorem 1 of [19]) that for a wide range of intuitionistic semantics, into which Kripke's can also be fit:

- If one can prove the completeness for the negative fragment of formulae (built using $\wedge, \bot, \Rightarrow, \forall$, and negated atomic formulae, $X \Rightarrow \bot$) then one can prove Markov's Principle. In view of Theorem 2.3, this implies that having a completeness proof cover $\bot$ means being able to prove Markov's Principle – which is known to be independent of many constructive logical systems, like Heyting Arithmetic or Constructive Type Theory.

- If one can prove the completeness for all connectives, i.e. including $\vee$ and $\exists$, then one can prove a strengthening[1] of the Double-negation Shift schema on Σ^0_1-formulae, which is also independent because it implies Markov's Principle.

We mentioned that Veldman [26] used Brouwer's Fan Theorem to handle $\vee$ and $\exists$, but to handle $\bot$ he included in his version of Kripke models an "exploding node" predicate, $\Vdash_\bot$ and defined $w \Vdash \bot := w \Vdash_\bot$. We remark in passing that Veldman's modification does not defy Kripke original definition, but only makes it more regular: if in Definition 2.1 one considers $\bot$ as an atomic formula, rather than a composite one, one falls back to Veldman's definition.

One can also try to straightforwardly extend the NBE-Completeness proof to cover disjunction (the existential quantifier is analogous) and see what happens. If one does that, one sees that a problem appears in the case of reflection of sum, $\uparrow^{A \vee B}$. There, given a neutral λ-term that derives $A \vee B$, one is supposed to prove that $w \Vdash A \vee B$ holds, which by definition means to prove that either $w \Vdash A$ or $w \Vdash B$ holds. But, since the input λ-term is neutral, it represents a blocked computation from which we will only be able to see whether A or B was derived, once we substitute values for the contained free variables that block the computation.

That is where the solution of Olivier Danvy appears. In [8], he used the full power[2] of the delimited control operators shift ($\mathcal{S}k.p$) and reset (#) [10] to give the following

[1] A special case of D-DNS$^+$ from page 13.

[2] We say "full power" because his usage of delimited control operators is strictly more powerful than what is possible with (non-delimited) control operators like call/cc. Danvy's program makes non-tail calls with continuations, while in the CPS translation of a program that uses call/cc all continuation calls are tail calls.

normalisation-by-evaluation algorithm for $\lambda^{\to\vee}$:

$$
\begin{array}{rll}
\downarrow^{\tau} : D \to \Lambda\text{-nf} & & \\
\downarrow^{\tau} := a \mapsto a & & \tau\text{-atomic} \\
\downarrow^{\tau\to\sigma} := S \mapsto \lambda a.\# \downarrow^{\sigma} (S \cdot \uparrow^{\tau} a) & & a\text{-fresh} \\
\downarrow^{\tau\vee\sigma} := S \mapsto \begin{cases} \iota_1(\downarrow^{\tau} S') & , \text{ if } S = \mathrm{inl} \cdot S' \\ \iota_2(\downarrow^{\sigma} S') & , \text{ if } S = \mathrm{inr} \cdot S' \end{cases} & & \\
\uparrow^{\tau} : \Lambda\text{-ne} \to D & & \\
\uparrow^{\tau} := a \mapsto a & & \tau\text{-atomic} \\
\uparrow^{\tau\to\sigma} := e \mapsto S \mapsto \uparrow^{\sigma} e(\downarrow^{\tau} S) & & \\
\uparrow^{\tau\vee\sigma} := e \mapsto \mathcal{S}\kappa.\mathsf{case}\ e\ \mathsf{of}\ (a_1.\#\kappa \cdot (\mathrm{inl} \cdot (\uparrow^{\tau} a_1)) \| a_2.\#\kappa \cdot (\mathrm{inr} \cdot (\uparrow^{\sigma} a_2))) & & a_i\text{-fresh}
\end{array}
$$

We characterise explicitly normal and neutral λ-terms by the following inductive definitions.

$$
\begin{array}{l}
\Lambda\text{-nf} \ni r := e^{\tau} \mid \lambda a^{\tau}.r^{\sigma} \mid \iota_1^{\tau} r \mid \iota_2^{\tau} r \\
\Lambda\text{-ne} \ni e := a^{\tau} \mid e^{\tau\to\sigma} r^{\tau} \mid \mathsf{case}\ e^{\tau\vee\sigma}\ \mathsf{of}\ \left(a_1^{\tau}.r_1^{\rho} \| a_2^{\sigma}.r_2^{\rho}\right)
\end{array}
$$

Given Danvy's NBE algorithm, which is simple and appears correct[3], does this mean that we can obtain a constructive proof of completeness for Kripke models if we permit delimited control operators in our ambient meta-language? Unfortunately, not, or not yet, because the available typing systems for them are either too complex (type-and-effect systems [10] change the meaning of implication), or do not permit to type-check the algorithm as a completeness proof (for example the typing system from [12], or the one from Chapter 4 of [17]).

3. Kripke-CPS Models and Their Completeness

However, there is a close connection between shift and reset, and the continuation-passing style (CPS) translations [11]. We can thus hope to give a normalisation-by-evaluation proof for full intuitionistic logic in continuation-passing style.

In this section we present a notion of model that we developed following this idea, by suitably inserting continuations into the notion of Kripke model. We prove that the new models are sound and complete for full intuitionistic predicate logic.

Definition 3.1. An *Intuitionistic Kripke-CPS model (IK-CPS)* is given by:

- a preorder $(K, \leq)$ of *possible worlds*;
- a binary relation on worlds $(-) \Vdash_{\perp}^{(-)}$ labelling a world as *exploding at a formula*;

[3] For more details on the computational behaviour of shift/reset and the algorithm itself, we refer the reader to the original paper [8] and to Section 3.2 of [17].

- a binary relation $(-) \Vdash_s (-)$ of *strong forcing* between worlds and atomic formulae, such that
$$\text{for all } w' \geq w, w \Vdash_s X \to w' \Vdash_s X,$$
- and a domain of quantification $D(w)$ for each world w, such that
$$\text{for all } w' \geq w, D(w) \subseteq D(w').$$

The relation $(-) \Vdash_s (-)$ of *strong forcing* is *extended from atomic to composite formulae* inductively and by simultaneously defining one new relation, (non-strong) forcing:

⋆ A formula A is *forced* in the world w (notation $w \Vdash A$) if, for any formula C,
$$\forall w' \geq w.\ \left(\forall w'' \geq w'.\ w'' \Vdash_s A \to w'' \Vdash_\bot^C\right) \to w' \Vdash_\bot^C;$$

- $w \Vdash_s A \wedge B$ if $w \Vdash A$ and $w \Vdash B$;
- $w \Vdash_s A \vee B$ if $w \Vdash A$ or $w \Vdash B$;
- $w \Vdash_s A \Rightarrow B$ if for all $w' \geq w$, $w \Vdash A$ implies $w \Vdash B$;
- $w \Vdash_s \forall x.A(x)$ if for all $w' \geq w$ and all $t \in D(w')$, $w' \Vdash A(t)$;
- $w \Vdash_s \exists x.A(x)$ if $w \Vdash A(t)$ for some $t \in D(w)$.

Remark 3.2. Certain details of the definition have been put into boxes to facilitate the comparison carried out in Section 5.

Lemma 3.3. *Strong forcing and (non-strong) forcing are monotone in any IK-CPS model, that is, given $w' \geq w$, $w \Vdash_s A$ implies $w' \Vdash_s A$, and $w \Vdash A$ implies $w' \Vdash A$.*

Proof. Monotonicity of strong forcing is proved by induction on the complexity of the formula, while that of forcing is by definition. The proof is easy and available in the Coq formalisation. □

Lemma 3.4. *The following monadic operations are definable for IK-CPS models:*

"unit" $\eta(\cdot)$ $\quad w \Vdash_s A \to w \Vdash A$

"bind" $(\cdot)^*(\cdot)$ $\quad (\forall w' \geq w.\ w' \Vdash_s A \to w' \Vdash B) \to w \Vdash A \to w \Vdash B$

Proof. Easy, using Lemma 3.3. If we leave implicit the handling of formulae C, worlds, and monotonicity, we have the following procedures behind the proofs.

$$\eta(\alpha) = \kappa \mapsto \kappa \cdot \alpha$$
$$(\phi)^*(\alpha) = \kappa \mapsto \alpha \cdot (\beta \mapsto \phi \cdot \beta \cdot \kappa)$$

□ □

$$\frac{(a : A) \in \Gamma}{\Gamma \vdash a : A}\ \mathrm{Ax}$$

$$\frac{\Gamma \vdash p : A_1 \qquad \Gamma \vdash q : A_2}{\Gamma \vdash (p, q) : A_1 \wedge A_2}\ \wedge_I \qquad \frac{\Gamma \vdash p : A_1 \wedge A_2}{\Gamma \vdash \pi_i p : A_i}\ \wedge^i_E$$

$$\frac{\Gamma \vdash p : A_i}{\Gamma \vdash \iota_i p : A_1 \vee A_2}\ \vee^i_I$$

$$\frac{\Gamma \vdash p : A_1 \vee A_2 \qquad \Gamma, a_1 : A_1 \vdash q_1 : C \qquad \Gamma, a_2 : A_2 \vdash q_2 : C}{\Gamma \vdash \mathsf{case}\ p\ \mathsf{of}\ (a_1.q_1 \| a_2.q_2) : C}\ \vee_E$$

$$\frac{\Gamma, a : A_1 \vdash p : A_2}{\Gamma \vdash \lambda a.p : A_1 \Rightarrow A_2}\ \Rightarrow_I \qquad \frac{\Gamma \vdash p : A_1 \Rightarrow A_2 \qquad \Gamma \vdash q : A_1}{\Gamma \vdash pq : A_2}\ \Rightarrow_E$$

$$\frac{\Gamma \vdash p : A(x) \qquad x\text{-fresh}}{\Gamma \vdash \lambda x.p : \forall x.A(x)}\ \forall_I \qquad \frac{\Gamma \vdash p : \forall x.A(x)}{\Gamma \vdash pt : A(t)}\ \forall_E$$

$$\frac{\Gamma \vdash p : A(t)}{\Gamma \vdash (t, p) : \exists x.A(x)}\ \exists_I$$

$$\frac{\Gamma \vdash p : \exists x.A(x) \qquad \Gamma, a : A(x) \vdash q : C \qquad x\text{-fresh}}{\Gamma \vdash \mathsf{dest}\ p\ \mathsf{as}\ (x.a)\ \mathsf{in}\ q : C}\ \exists_E$$

Table 1: Proof term annotation for the natural deduction system of minimal intuitionistic predicate logic (MQC)

With Table 1, we fix a derivation system and proof term notation for minimal intuitionistic predicate logic. There are two kinds of variables, proof term variables $a, b, \ldots$ and individual (quantifier) variables $x, y, \ldots$. Individual constants are denoted by t. We rely on these conventions to resolve the apparent ambiguity of the syntax: the abstraction $\lambda a.p$ is a proof term for implication, while $\lambda x.p$ is a proof term for $\forall$; (p, q) is a proof term for $\wedge$, while (t, q) is a proof term for $\exists$.

We supplement the characterisation of normal and neutral terms from page 5:

$$\begin{aligned}
&\Lambda\text{-nf} \ni r ::= e \mid \lambda a.r \mid \iota_1 r \mid \iota_2 r \mid (r_1, r_2) \mid \lambda x.r \mid (t, r)\\
&\Lambda\text{-ne} \ni e ::= a \mid er \mid \mathsf{case}\ e\ \mathsf{of}\ (a_1.r_1 \| a_2.r_2) \mid \pi_1 e \mid \pi_2 e \mid et \mid\\
&\qquad\qquad \mathsf{dest}\ e\ \mathsf{as}\ (x.a)\ \mathsf{in}\ r
\end{aligned}$$

As before, let $w \Vdash \Gamma$ denote that all formulae from Γ are forced.

Theorem 3.5 (Soundness)**.** *If $\Gamma \vdash p : A$, then, in any world w of any IK-CPS model, if $w \Vdash \Gamma$, then $w \Vdash A$.*

Proof. This is proved by a simple induction on the length of the derivation. We give the algorithm behind it in section 4. □

Remark 3.6. The condition "for all formula C" in Definition 3.1 is only necessary for the soundness proof to go through, more precisely, the cases of elimination rules for $\vee$ and $\Rightarrow$. The completeness proof goes through even if we define forcing by

$$\forall w' \geq w.\ \left(\forall w'' \geq w'.\ w'' \Vdash_s A \to w'' \Vdash_\perp^A\right) \to w' \Vdash_\perp^A .$$

Definition 3.7. The *Universal IK-CPS model* $\mathcal{U}$ is obtained by setting:

- K to be the set of contexts Γ of MQC;
- $\Gamma \leq \Gamma'$ iff $\Gamma \subseteq \Gamma'$;
- $\Gamma \Vdash_s X$ iff there is a derivation in normal form of $\Gamma \vdash X$ in MQC, where X is an atomic formula;
- $\Gamma \Vdash_\perp^C$ iff there is a derivation in normal form of $\Gamma \vdash C$ in MQC;
- for any w, $D(w)$ is a set of individuals for MQC (that is, $D(-)$ is a constant function from worlds to sets of individuals).

$(-) \Vdash_s (-)$ is monotone because of the weakening property for intuitionistic "$\vdash$".

Remark 3.8. The difference between strong forcing "$\Vdash_s$" and the exploding node predicate "$\Vdash_\perp^C$" in $\mathcal{U}$ is that the former is defined on atomic formulae, while the latter is defined on any kind of formulae.

Lemma 3.9. *We can also define the monadic* "run" *operation on the universal model $\mathcal{U}$,* for atomic formulae *X:*

$$\mu(\cdot) : w \Vdash X \to w \Vdash_s X.$$

Proof. By setting $C := A$ and applying the identity function. □

Theorem 3.10 (Completeness for $\mathcal{U}$)**.** *For any closed formula A and closed context Γ, the following hold for $\mathcal{U}$:*

$$\Gamma \Vdash A \longrightarrow \{p \mid \Gamma \vdash p : A\} \qquad (\textit{"reify"}) \qquad (\downarrow)$$

$$\Gamma \vdash e : A \longrightarrow \Gamma \Vdash A \qquad (\textit{"reflect"}) \qquad (\uparrow)$$

Moreover, the target of ($\downarrow$) is a normal term, while the source of ($\uparrow$) is a neutral term.

Proof. We prove simultaneously the two statements by induction on the complexity of formula A.

We skip writing the proof term annotations, and write just $\Gamma \vdash A$ instead of "there exists p such that $\Gamma \vdash p : A$", in order to decrease the level of detail. The algorithm behind this proof that concentrates on proof terms is given in Section 4.

Base case. ($\downarrow$) is by "run" (Lemma 3.9), ($\uparrow$) is by "unit" (Lemma 3.4).

Induction case for $\wedge$. Let $\Gamma \Vdash A \wedge B$ i.e.

$$\forall C.\ \forall \Gamma' \geq \Gamma.\ ((\forall \Gamma'' \geq \Gamma'.\ \Gamma'' \Vdash A \text{ and } \Gamma'' \Vdash B \to \Gamma'' \vdash C) \to \Gamma' \vdash C)\,.$$

We apply this hypothesis by setting $C := A \wedge B$ and $\Gamma' := \Gamma$, and then, given $\Gamma'' \geq \Gamma$ s.t. $\Gamma'' \Vdash A$ and $\Gamma'' \Vdash B$, we have to derive $\Gamma'' \vdash A \wedge B$. But, this is immediate by applying the $\wedge_I$ rule and the induction hypothesis ($\downarrow$) twice, for A and for B.

Let $\Gamma \vdash A \wedge B$ be a neutral derivation. We prove $\Gamma \Vdash A \wedge B$ by applying unit (Lemma 3.4), and then applying the induction hypothesis ($\downarrow$) on $\wedge_I^1$, $\wedge_I^2$, and the hypothesis.

Induction case for $\vee$. Let $\Gamma \Vdash A \vee B$ i.e.

$$\forall C.\ \forall \Gamma' \geq \Gamma.\ ((\forall \Gamma'' \geq \Gamma'.\ \Gamma'' \Vdash A \text{ or } \Gamma'' \Vdash B \to \Gamma'' \vdash C) \to \Gamma' \vdash C)\,.$$

We apply this hypothesis by setting $C := A \vee B$ and $\Gamma' := \Gamma$, and then, given $\Gamma'' \geq \Gamma$ s.t. $\Gamma'' \Vdash A$ or $\Gamma'' \Vdash B$, we have to derive $\Gamma'' \vdash A \vee B$. But, this is immediate, after a case distinction, by applying the $\vee_I^i$ rule and the induction hypothesis ($\downarrow$).

We now consider the only case (besides $\uparrow^{\exists x A(x)}$ below) where using shift and reset, or our Kripke-style models, is crucial. Let $\Gamma \vdash A \vee B$ be a neutral derivation. Let a formula C and $\Gamma' \geq \Gamma$ be given, and let

$$\forall \Gamma'' \geq \Gamma'.\ (\Gamma'' \Vdash A \text{ or } \Gamma'' \Vdash B \to \Gamma'' \vdash C)\,. \tag{\#}$$

We prove $\Gamma' \vdash C$ by the following derivation tree:

$$\dfrac{\dfrac{\Gamma \vdash A \vee B}{\Gamma' \vdash A \vee B} \qquad \dfrac{\dfrac{\dfrac{\dfrac{A \in A, \Gamma'}{A, \Gamma' \vdash A}\,\text{Ax}}{A, \Gamma' \Vdash A}\,(\uparrow)}{A, \Gamma' \Vdash A \text{ or } A, \Gamma' \Vdash B}\,\text{inl}}{A, \Gamma' \vdash C}\,(\#) \qquad \dfrac{\dfrac{\dfrac{\dfrac{B \in B, \Gamma'}{B, \Gamma' \vdash B}\,\text{Ax}}{B, \Gamma' \Vdash B}\,(\uparrow)}{B, \Gamma' \Vdash A \text{ or } B, \Gamma' \Vdash B}\,\text{inr}}{B, \Gamma' \vdash C}\,(\#)}{\Gamma' \vdash C}\,\vee_E$$

Induction case for $\Rightarrow$. Let $\Gamma \Vdash A \Rightarrow B$ i.e.

$$\forall C.\ \forall \Gamma' \geq \Gamma.\ ((\forall \Gamma'' \geq \Gamma'.\ (\forall \Gamma_3 \geq \Gamma''.\ \Gamma_3 \Vdash A \to \Gamma_3 \Vdash B) \to \Gamma'' \vdash C) \to \Gamma' \vdash C)\,.$$

We apply this hypothesis by setting $C := A \Rightarrow B$ and $\Gamma' := \Gamma$, and then, given $\Gamma'' \geq \Gamma$ s.t.

$$\forall \Gamma_3 \geq \Gamma''.\ \Gamma_3 \Vdash A \to \Gamma_3 \Vdash B \tag{\#}$$

we have to derive $\Gamma'' \vdash A \Rightarrow B$. This follows by applying ($\Rightarrow_I$), the IH for($\downarrow$), then (#), and finally the IH for ($\uparrow$) with the Ax rule.

Let $\Gamma \vdash A \Rightarrow B$ be a neutral derivation. We prove $\Gamma \Vdash A \Rightarrow B$ by applying unit (Lemma 3.4), and then, given $\Gamma' \geq \Gamma$ and $\Gamma' \Vdash A$, we have to show that $\Gamma' \Vdash B$. This is done by applying the IH for ($\uparrow$) on the ($\Rightarrow_E$) rule, with the IH for ($\downarrow$) applied to $\Gamma' \Vdash A$.

Induction case for $\forall$. We recall that the domain function $D(-)$ is constant in the universal model $\mathcal{U}$. Let $\Gamma \Vdash \forall x A(x)$ i.e.

$$\forall C.\, \forall \Gamma' \geq \Gamma.\ ((\forall \Gamma'' \geq \Gamma'.\ (\forall \Gamma_3 \geq \Gamma''.\, \forall t \in D.\, \Gamma_3 \Vdash A(t)) \to \Gamma'' \vdash C) \to \Gamma' \vdash C)\,.$$

We apply this hypothesis by setting $C := \forall x A(x)$ and $\Gamma' := \Gamma$, and then, given $\Gamma'' \geq \Gamma$ s.t.

$$\forall \Gamma_3 \geq \Gamma''.\, \forall t \in D.\, \Gamma_3 \Vdash A(t) \tag{\#}$$

we have to derive $\Gamma'' \vdash \forall x A(x)$. This follows by applying $(\forall_I)$, the IH for$(\downarrow)$, and then (#).

Let $\Gamma \vdash \forall x A(x)$ be a neutral derivation. We prove $\Gamma \Vdash \forall x A(x)$ by applying unit (Lemma 3.4), and then, given $\Gamma' \geq \Gamma$ and $t \in D$, we have to show that $\Gamma' \Vdash A(t)$. This is done by applying the IH for $(\uparrow)$ on the $(\forall_E)$ rule and the hypothesis $\Gamma \vdash \forall x A(x)$.

Induction case for $\exists$. Let $\Gamma \Vdash \exists x A(x)$ i.e.

$$\forall C.\, \forall \Gamma' \geq \Gamma.\ ((\forall \Gamma'' \geq \Gamma'.\ (\exists t \in D.\, \Gamma'' \Vdash A(t)) \to \Gamma'' \vdash C) \to \Gamma' \vdash C)\,.$$

We apply this hypothesis by setting $C := \exists x A(x)$ and $\Gamma' := \Gamma$, and then, given $\Gamma'' \geq \Gamma$ s.t. $\exists t \in D.\, \Gamma'' \Vdash A(t)$, we have to derive $\Gamma'' \vdash \exists x A(x)$. This follows by applying $(\exists_I)$ with $t \in D$, and the IH for$(\downarrow)$.

Let $\Gamma \vdash \exists x A(x)$ be a neutral derivation. Let a formula C and $\Gamma' \geq \Gamma$ be given, and let

$$\forall \Gamma'' \geq \Gamma'.\ (\exists t \in D. \Gamma'' \Vdash A(t) \to \Gamma'' \vdash C)\,. \tag{\#}$$

We prove $\Gamma' \vdash C$ by the following derivation tree:

$$\dfrac{\dfrac{\Gamma \vdash \exists x A(x)}{\Gamma' \vdash \exists x A(x)} \qquad \dfrac{\dfrac{\dfrac{\dfrac{A(x) \in A(x), \Gamma'}{A(x), \Gamma' \vdash A(x)}\,\text{Ax}}{A(x), \Gamma' \Vdash A(x)}\,(\uparrow)}{A(x), \Gamma' \vdash C}\,(\#)}{\Gamma' \vdash C}\ x\text{-fresh}\ \exists_E$$

The result of reification "$\downarrow$*" is in normal form.* By inspection of the proof. □ □

4. Normalisation by Evaluation in IK-CPS Models

In this section we give the algorithm that we manually extracted from the Coq formalisation, for the restriction to the interesting propositional fragment that involves implication and disjunction. The algorithm extracted automatically by Coq contains too many details to be instructive.

The following evaluation function for $\lambda^{\to\vee}$-terms is behind the proof of Theorem 3.5:

$$[\![\Gamma \vdash p : A]\!]_{w \Vdash \Gamma} : w \Vdash A$$

$$\begin{aligned}
[\![a]\!]_\rho &:= \rho(a) \\
[\![\lambda a.p]\!]_\rho &:= \kappa \mapsto \kappa \cdot \left(\alpha \mapsto [\![p]\!]_{\rho, a \mapsto \alpha}\right) = \eta \cdot \left(\alpha \mapsto [\![p]\!]_{\rho, a \mapsto \alpha}\right) \\
[\![pq]\!]_\rho &:= \kappa \mapsto [\![p]\!]_\rho \cdot \left(\phi \mapsto \phi \cdot [\![q]\!]_\rho \cdot \kappa\right) \\
[\![\iota_1 p]\!]_\rho &:= \kappa \mapsto \kappa \cdot \left(\mathrm{inl} \cdot [\![p]\!]_\rho\right) = \eta \cdot \left(\mathrm{inl} \cdot [\![p]\!]_\rho\right) \\
[\![\iota_2 p]\!]_\rho &:= \kappa \mapsto \kappa \cdot \left(\mathrm{inr} \cdot [\![p]\!]_\rho\right) = \eta \cdot \left(\mathrm{inr} \cdot [\![p]\!]_\rho\right) \\
[\![\mathsf{case}\ p\ \mathsf{of}\ (a_1.q_1 \| a_2.q_2)]\!]_\rho &:= \kappa \mapsto [\![p]\!]_\rho \cdot \left(\gamma \mapsto \begin{cases} [\![q_1]\!]_{\rho, a_1 \mapsto \alpha} \cdot \kappa & \text{if } \gamma = \mathrm{inl} \cdot \alpha \\ [\![q_2]\!]_{\rho, a_2 \mapsto \beta} \cdot \kappa & \text{if } \gamma = \mathrm{inr} \cdot \beta \end{cases}\right)
\end{aligned}$$

The following is the algorithm behind Theorem 3.10:

$$\begin{aligned}
&\downarrow^A_\Gamma : \Gamma \Vdash A \to \{p \in \Lambda\text{-nf} \mid \Gamma \vdash p : A\} \\
&\uparrow^A_\Gamma : \{e \in \Lambda\text{-ne} \mid \Gamma \vdash e : A\} \to \Gamma \Vdash A
\end{aligned}$$

$$\begin{aligned}
\downarrow^X_\Gamma &:= \alpha \mapsto \mu \cdot \alpha && X\text{-atomic} \\
\uparrow^X_\Gamma &:= e \mapsto \eta \cdot e && X\text{-atomic} \\
\downarrow^{A \Rightarrow B}_\Gamma &:= \eta \cdot \left(\phi \mapsto \lambda a.\ \downarrow^B_{\Gamma, a:A} \left(\phi \cdot \uparrow^A_{\Gamma, a:A} a\right)\right) && a\text{-fresh} \\
\uparrow^{A \Rightarrow B}_\Gamma &:= e \mapsto \eta \cdot \left(\alpha \mapsto \uparrow^B_\Gamma \left(e \left(\downarrow^A_\Gamma \alpha\right)\right)\right) \\
\downarrow^{A \vee B}_\Gamma &:= \eta \cdot \left(\gamma \mapsto \begin{cases} \iota_1 \downarrow^A_\Gamma \alpha & \text{if } \gamma = \mathrm{inl} \cdot \alpha \\ \iota_2 \downarrow^B_\Gamma \beta & \text{if } \gamma = \mathrm{inr} \cdot \beta \end{cases}\right) \\
\uparrow^{A \vee B}_\Gamma &:= e \mapsto \kappa \mapsto \mathsf{case}\ e\ \mathsf{of}\ \left(a_1.\kappa \cdot \left(\mathrm{inl} \cdot \uparrow^A_{\Gamma, a_1:A} a_1\right) \| a_2.\kappa \cdot \left(\mathrm{inr} \cdot \uparrow^B_{\Gamma, a_2:B} a_2\right)\right) && a_i\text{-fresh}
\end{aligned}$$

5. Variants and Relation to Kripke Models

5.1. "Call-by-value" Models

Defining forcing on composite formulae in Definition 3.1 proceeds analogously to defining the call-by-name CPS translation [23], or Kolmogorov's double-negation translation [25, 22]. A definition analogous to the "call-by-value" CPS translation [23] is also possible, by defining (non-strong) forcing by:

- $w \Vdash_{\mathrm{s}} A \wedge B$ if $w \Vdash_{\mathrm{s}} A$ and $w \Vdash_{\mathrm{s}} B$;
- $w \Vdash_{\mathrm{s}} A \vee B$ if $w \Vdash_{\mathrm{s}} A$ or $w \Vdash_{\mathrm{s}} B$;
- $w \Vdash_{\mathrm{s}} A \Rightarrow B$ if for all $w' \geq w$, $w \Vdash_{\mathrm{s}} A$ implies $w \Vdash B$;
- $w \Vdash_{\mathrm{s}} \forall x.A(x)$ if for all $w' \geq w$ and all $t \in D(w')$, $w' \Vdash A(t)$;

- $w \Vdash_{\mathrm{s}} \exists x.A(x)$ if $w \Vdash_{\mathrm{s}} A(t)$ for some $t \in D(w)$.

One can prove this variant of IK-CPS models sound and complete, similarly to Section 3, except for two differences. Firstly, in the statement of Soundness, one needs to put $w \Vdash_{\mathrm{s}} \Gamma$ in place of $w \Vdash \Gamma$. Secondly, due to the first difference, the composition of soundness of completeness that gives normalisation works for *closed* terms only.

5.2. Classical Models

In [16, 17, 18], we presented the following notion of model which is complete for *classical* predicate logic and represents an NBE algorithm for it.

Definition 5.1. A *Classical Kripke-CPS model (CK-CPS)*, is given by:

- a preorder $(K, \leq)$ of *possible worlds*;
- a unary relation on worlds $(-) \Vdash_{\bot}$ labelling a world as *exploding*;
- a binary relation $(-) \Vdash_{\mathrm{s}} (-)$ of *strong forcing* between worlds and atomic formulae, such that
$$\text{for all } w' \geq w, w \Vdash_{\mathrm{s}} X \to w' \Vdash_{\mathrm{s}} X,$$
- and a domain of quantification $D(w)$ for each world w, such that
$$\text{for all } w' \geq w, D(w) \subseteq D(w').$$

The relation $(-) \Vdash_{\mathrm{s}} (-)$ of *strong forcing* is *extended from atomic to composite formulae* inductively and by simultaneously defining two new relations, refutation and (non-strong) forcing:

- ⋆ A formula A is *refuted* in the world w (notation $w : A \Vdash$) if any world $w' \geq w$, which strongly forces A, is exploding;
- ⋆ A formula A is *forced* in the world w (notation $w \Vdash A$) if any world $w' \geq w$, which refutes A, is exploding;
- $w \Vdash_{\mathrm{s}} A \wedge B$ if $w \Vdash A$ and $w \Vdash B$;
- $w \Vdash_{\mathrm{s}} A \vee B$ if $w \Vdash A$ or $w \Vdash B$;
- $w \Vdash_{\mathrm{s}} A \Rightarrow B$ if for all $w' \geq w$, $w \Vdash A$ implies $w \Vdash B$;
- $w \Vdash_{\mathrm{s}} \forall x.A(x)$ if for all $w' \geq w$ and all $t \in D(w')$, $w' \Vdash A(t)$;
- $w \Vdash_{\mathrm{s}} \exists x.A(x)$ if $w \Vdash A(t)$ for some $t \in D(w)$.

The differences between Definition 3.1 and Definition 5.1 are marked with boxes. We can also present CK-CPS using binary exploding nodes, by defining $w \Vdash_{\mathrm{s}} \bot := \forall C.w \Vdash^{C}_{\bot}$. Then, we get the following statement of forcing in CK-CPS,

$$\forall w' \geq w.\ \left(\forall w'' \geq w'.\ w'' \Vdash_{\mathrm{s}} A \to \forall I.w'' \Vdash^{I}_{\bot}\right) \to \forall O.w' \Vdash^{O}_{\bot},$$

versus forcing in IK-CPS,

$$\forall C.\, \forall w' \geq w.\, \left(\forall w'' \geq w'.\, w'' \Vdash_{\mathrm{s}} A \to w'' \Vdash_{\bot}^{C}\right) \to w' \Vdash_{\bot}^{C}.$$

The difference between forcing in the intuitionistic and classical models is, then, that: 1) the dependency on C is necessary in the intuitionistic case, while it is optional in the classical case; 2) the continuation (the internal implication) in classical forcing is allowed to change the parameter C upon application, whereas in intuitionistic forcing the parameter is not local to the continuation, but to the continuation of the continuation.

At this point we also remark that the use of dependent types to handle the parameter C is determined by the fact that we formalise our definitions in Intuitionistic Type Theory. Otherwise, the quantification $\forall C.\cdots$ is quantification over first-order individuals, for example natural numbers.

5.3. Kripke Models

Let $A(n)$ be an arbitrary first-order formula and let $X(n,m)$ be a Σ_1^0-formula. Denote the following arithmetic schema by (D-DNS$^+$) for "dependent Double-negation Shift schema, strengthened".

$$\frac{\forall m.\, \forall n_1 \geq n.\, (\forall n_2 \geq n_1.\, A(n_2) \to X(n_2,m)) \to X(n_1,m)}{A(n)}\ \text{D-DNS}^+$$

Proposition 5.2. *Let $\mathcal{K} = (K, \leq, D, \vDash, \vDash_{\bot})$ be any structure such that $\vDash$ denotes forcing in the standard Kripke model arising from $\mathcal{K}$, and $\Vdash$ denotes (non-strong) forcing in the IK-CPS model arising from the same $\mathcal{K}$.*

Then, in the presence of (D-DNS$^+$) at meta-level, for all formula A, and any $w \in K$,

$$w \vDash A \longleftrightarrow w \Vdash A.$$

Proof. The proof is by induction on A, using (D-DNS$^+$) to prove,

$$\frac{\forall C.\, \forall w_1 \geq w.\, \left(\forall w_2 \geq w_1.\, (w_2 \Vdash A \text{ or } w_2 \Vdash B) \to w_2 \Vdash_{\bot}^{C})\right) \to w_1 \Vdash_{\bot}^{C}}{w \Vdash A \text{ or } w \Vdash B},$$

needed in the case for disjunction, and similarly for the existential quantifier. □

Corollary 5.3. *Completeness of full intuitionistic predicate logic with respect to standard Kripke models is provable constructively, in the presence of D-DNS$^+$.*

Remark 5.4. It is the other direction of this implication that Kreisel proved, for a specialisation of D-DNS$^+$. (Section 2) To investigate more precisely whether D-DNS$^+$ captures exactly constructive provability of completeness for Kripke models remains future work.

6. Conclusion

We emphasised that our algorithm is β-NBE, because were we able to identify $\beta\eta$-equal terms of $\lambda^{\to\vee}$ through our NBE function, we would have solved the problem of the existence of canonical η-long normal form for $\lambda^{\to\vee}$. However, as shown by [14], due to the connection with Tarski's High School Algebra Problem [5, 27], the notion of such a normal form is not finitely axiomatisable. If one looks at examples of $\lambda^{\to\vee}$-terms which are $\beta\eta$-equal but are not normalised to the same term by Danvy's (and our) algorithm, one can see that in the Coq type theory these terms are interpreted as denotations that involve commutative cuts.

In recent unpublished work [9], Danvy also developed a version of his NBE algorithm directly in CPS, without using delimited control operators.

In [2], Barral gives a program for NBE of λ-calculus with sums by just using the exceptions mechanism of a programming language, which is something *a priori* strictly weaker than using delimited control operators.

In [1], Altenkirch, Dybjer, Hofmann, and Scott, give a topos theoretic proof of NBE for a typed λ-calculus with sums, by constructing a sheaf model. The connection between sheaves and Beth semantics[4] is well known. While the proof is constructive, due to their use of topos theory, we were unable to extract an algorithm from it.

In [21], Macedonio and Sambin present a notion of model for extensions of Basic logic (a sub-structural logic more primitive than Linear logic), which, for intuitionistic logic, appears to be related to our notion of model. However, they demand that their set of worlds K be saturated, while we do not, and we can hence also work with finite models.

In [13], Filinski proves the correctness of an NBE algorithm for Moggi's computational λ-calculus, including sums. We found out about Filinski's paper right before finishing our own. He also evaluates the input terms in a domain based on continuations.

Acknowledgements

To Hugo Herbelin for inspiring discussions and, in particular, for suggesting to try polymorphism, viz. Remark 3.6. To Olivier Danvy for suggesting reference [13], and for pioneering work on delimited control operators.

References

[1] Thorsten Altenkirch, Peter Dybjer, Martin Hofmann, and Philip J. Scott. Normalization by evaluation for typed lambda calculus with coproducts. In *LICS*, pages 303–310, 2001.

[4] We remark that, for the fragment $\{\Rightarrow, \forall, \wedge\}$, NBE can also be seen as completeness for *Beth* semantics, since forcing in Beth and Kripke models is the same thing on that fragment.

[2] Freiric Barral. Exceptional NbE for sums. In Olivier Danvy, editor, *Informal proceedings of the 2009 Workshop on Normalization by Evaluation, August 15th 2009, Los Angeles, California*, pages 21–30, 2009.

[3] U. Berger and P. Oliva. Modified bar recursion and classical dependent choice. In M. Baaz, S.D. Friedman, and J. Kraijcek, editors, *Logic Colloquium '01, Proceedings of the Annual European Summer Meeting of the Association for Symbolic Logic, held in Vienna, Austria, August 6 - 11, 2001*, volume 20 of *Lecture Notes in Logic*, pages 89–107. Springer, 2005.

[4] Ulrich Berger and Helmut Schwichtenberg. An inverse of the evaluation functional for typed lambda-calculus. In *LICS*, pages 203–211. IEEE Computer Society, 1991.

[5] Stanley Burris and Simon Lee. Tarski's high school identities. *The American Mathematical Monthly*, 100(3):231–236, 1993.

[6] Catarina Coquand. From semantics to rules: A machine assisted analysis. In *CSL '93*, volume 832 of *Lecture Notes in Computer Science*, pages 91–105. Springer, 1993.

[7] Catarina Coquand. A formalised proof of the soundness and completeness of a simply typed lambda-calculus with explicit substitutions. *Higher Order Symbol. Comput.*, 15(1):57–90, 2002.

[8] Olivier Danvy. Type-directed partial evaluation. In *POPL*, pages 242–257, 1996.

[9] Olivier Danvy. A call-by-name normalization function for the simply typed lambda-calculus with sums and products. manuscript, 2008.

[10] Olivier Danvy and Andrzej Filinski. A functional abstraction of typed contexts. Technical report, Computer Science Department, University of Copenhagen, 1989. DIKU Rapport 89/12.

[11] Olivier Danvy and Andrzej Filinski. Abstracting control. In *LISP and Functional Programming*, pages 151–160, 1990.

[12] Andrzej Filinski. *Controlling Effects*. PhD thesis, School of Computer Science, Carnegie Mellon University, 1996. Technical Report CMU-CS-96-119 (144pp.).

[13] Andrzej Filinski. Normalization by evaluation for the computational lambda-calculus. In Samson Abramsky, editor, *Typed Lambda Calculi and Applications*, volume 2044 of *Lecture Notes in Computer Science*, pages 151–165. Springer Berlin / Heidelberg, 2001.

[14] Marcelo P. Fiore, Roberto Di Cosmo, and Vincent Balat. Remarks on isomorphisms in typed lambda calculi with empty and sum types. *Ann. Pure Appl. Logic*, 141(1-2):35–50, 2006.

[15] Hugo Herbelin and Gyesik Lee. Forcing-based cut-elimination for Gentzen-style intuitionistic sequent calculus. In Hiroakira Ono, Makoto Kanazawa, and Ruy J. G. B. de Queiroz, editors, *WoLLIC*, volume 5514 of *Lecture Notes in Computer Science*, pages 209–217. Springer, 2009.

[16] Danko Ilik. Formalisation of completeness for Kripke-CPS models, 2009. `http://www.lix.polytechnique.fr/~danko/code/kripke_completeness/`.

[17] Danko Ilik. *Constructive Completeness Proofs and Delimited Control*. PhD thesis, École Polytechnique, October 2010.

[18] Danko Ilik, Gyesik Lee, and Hugo Herbelin. Kripke models for classical logic. *Annals of Pure and Applied Logic*, 161(11):1367 – 1378, 2010. Special Issue: Classical Logic and Computation (2008).

[19] Georg Kreisel. On weak completeness of intuitionistic predicate logic. *J. Symb. Log.*, 27(2):139–158, 1962.

[20] Saul Kripke. Semantical considerations on modal and intuitionistic logic. *Acta Philos. Fennica*, 16:83–94, 1963.

[21] Damiano Macedonio and Giovanni Sambin. From meta-level to semantics via reflection: a model for basic logic and its extensions. available from the authors.

[22] Chetan Murthy. *Extracting Classical Content from Classical Proofs*. PhD thesis, Department of Computer Science, Cornell University, 1990.

[23] G. D. Plotkin. Call-by-name, call-by-value and the [lambda]-calculus. *Theoretical Computer Science*, 1(2):125–159, 1975.

[24] Clifford Spector. Provably recursive functionals of analysis: a consistency proof of analysis by an extension of principles formulated in current intuitionistic mathematics. In *Proc. Sympos. Pure Math., Vol. V*, pages 1–27. American Mathematical Society, Providence, R.I., 1962.

[25] A. S. Troelstra and D. van Dalen. *Constructivism in mathematics. Vol. I*, volume 121 of *Studies in Logic and the Foundations of Mathematics*. North-Holland Publishing Co., Amsterdam, 1988. An introduction.

[26] Wim Veldman. An intuitionistic completeness theorem for intuitionistic predicate logic. *J. Symb. Log.*, 41(1):159–166, 1976.

[27] A. J. Wilkie. On exponentiation - a solution to Tarski's high school algebra problem. Technical report, Mathematical Institute, Oxford, UK, 2001.