The Typed Böhm Theorem

Kosta Došen
University of Toulouse III, IRIT
31062 Toulouse cedex, France and Mathematical Institute
Knez Mihailova 35, P.O. Box 367
11001 Belgrade, Yugoslavia
email: kosta@mi.sanu.ac.yu

Zoran Petrić
Mathematical Institute
Knez Mihailova 35, P.O. Box 367
11001 Belgrade, Yugoslavia
email: zpetric@mi.sanu.ac.yu

Abstract

A new proof is given of the analogue of Böhm's Theorem in the typed lambda calculus with functional types. This note is a summary of: K. Došen and Z. Petrić, "The maximality of the typed lambda calculus and of cartesian closed categories" (*Publications de l'Institut Mathématique (N.S.)* 68(82) (2000), pp. 1-19; http:// arXiv. org/ math. CT/ 9911073).

Mathematics Subject Classification (2000): 03B40

1 Introduction

We give a new proof of the analogue of Böhm's Theorem in the typed lambda calculus with only functional types. This result was already established in [11] (Theorem 2), without mentioning Böhm's Theorem. Statman has even a semantic notion of consistent extension, rather than a syntactic notion, such as we have, following Böhm. (The two notions happen to be equivalent, however.) Our analogue of Böhm's Theorem in the typed lambda calculus is closer to standard formulations of this theorem, and our proof is different from Statman's, which relies on the type-reducing result of [10] (Theorem 3). Our approach provides an alternative proof of this type-reducing result. We rely on a different result from the same paper [10] (Theorem 2), proved previously in [9] (Theorem 2), which is a finite-model property for the typed lambda calculus. There are, however, some analogies in the general spirit of these proofs. In order to use the finite model property for the typed lambda calculus, we define the sets of P-functionals starting from a finite set P, with the help of exponentiation (B^A is the set of all functions from A to B). We show that they are all lambda-definable in the sense that for two lambda terms a and b whose interpretations in a finite model based on P are not equal, there is a syntactical procedure deriving $[m] = [n]$ from some type instance $a = b$, where $[m]$ and $[n]$ are Church numerals for $m \neq n$. We hope that our method might shed some new light on the matter.

The possibility of proceeding as we do is indicated briefly in [8] (last paragraph of section 5). Simpson says: "It is an interesting fact that an alternative direct proof of Theorem 3 is possible using a typed version of the Böhm-out technique [1] (Chapter 10). The details are beyond the scope of this paper." We don't know what Böhm-out technique Simpson had in mind, but he assured us his approach is different from ours. Anyway, we couldn't find such a technique by imitating [1]. Our technique has some intrinsic difficulties, but presumably not more than the technique of [11]. Our presentation takes a little bit more space because we have tried to help the reader by going into more details. These details, which were beyond the scope of Simpson's paper, fall exactly within the scope of ours.

2 Böhm's Theorem

Böhm's Theorem in the untyped lambda calculus says that if a and b are two different lambda terms in $\beta\eta$ normal form, and c and d are arbitrary lambda terms, then one can construct terms $h_1, \ldots, h_n$, $n \geq 0$, and find variables $x_1, \ldots, x_m$, $m \geq 0$, such that

$$(\lambda_{x_1 \ldots x_m} a) h_1 \ldots h_n = c,$$

$$(\lambda_{x_1 \ldots x_m} b) h_1 \ldots h_n = d$$

are provable in the β lambda calculus (see [1], Chapter 10, §4, Theorem 10.4.2; [4], Chapter 11F, §8, Theorem 5; [6], Chapitre V, Théorème 2; we know the original paper of Böhm [3] only from references). As a corollary of this theorem one obtains that if a and b are two lambda terms having a normal form such that $a = b$ is not provable in the $\beta\eta$ lambda calculus and this calculus is extended with $a = b$, then one can prove every equality in the extended calculus.

Here we demonstrate the analogue of Böhm's Theorem in the typed lambda calculus with only functional types. The standard proof of Böhm's Theorem, which may be found for example in [1], cannot be transferred to the typed case. At crucial places it introduces lambda terms that cannot be appropriately typed. For example, for $\lambda_{xy} xy$ and $\lambda_{xy} x(xy)$ (i.e., the Church numerals for 1 and 2) with x of type $p \to p$ and y of type p there is no appropriate permutator of type $p \to p$ with whose help these two terms can be transformed into terms with a head original head normal form (see [1], Chapter 10, §3). A more involved example is given with the terms $\lambda_x x \lambda_y (x \lambda_z y)$ and $\lambda_x x \lambda_y (x \lambda_z z)$ with x of type $(p \to p) \to p$ and y and z of type p (we deal with these two typed terms in the Example of Section 6).

One cannot deduce our analogue of Böhm's Theorem for the typed lambda calculus from Böhm's Theorem for the untyped lambda calculus. The typed calculus has a more restricted language and does not allow everything permitted in the untyped case. Conversely, one cannot deduce Böhm's Theorem for the untyped lambda calculus from our typed version of this theorem. Our result covers only cases where a and b are typable by the same type.

3 The typed lambda calculus

The formulation of the typed lambda calculus with only functional types we rely on is rather standard (see, for example, [1], Appendix 1, or [5]). However, we sketch this formulation briefly, to fix notation and terminology.

Types are defined inductively by a nonempty set of *atomic types* and the clause "if A and B are types, then $(A \to B)$ is a type". For atomic types we use the schematic letters p, q, r, …, p_1, …, and for all types we use the schematic letters A, B, C, …, A_1, … We write A^p_B for the result of substituting B for p in A. (*Substitution* means as usual *uniform replacement.*)

Terms are defined inductively in a standard manner. We have infinitely many *variables* of each type, for which we use the schematic letters x, y, z, …, x_1, … For arbitrary terms we use the schematic letters a, b, c, …, a_1, … That a term a is of type A is expressed by $a : A$. However, for easier reading, we will not write types inside terms, but will specify the types of variables separately. For application we use the standard notation, with the standard omitting of parentheses. For lambda abstraction we will write λ_x with subscripted x, instead of λx (this way we can do without dots in $\lambda_x x$, which is otherwise written $\lambda x.x$). We abbreviate $\lambda_{x_1} \ldots \lambda_{x_n} a$ by $\lambda_{x_1 \ldots x_n} a$, as usual. We write a^x_b for the result of substituting b for x in a, provided b is free for x in a.

If a is a term, let a *type-instance* of a be obtained by substituting some types for the atomic types in the variables of a.

A *formula* of the typed lambda calculus Λ is of the form $a = b$ where a and b are terms of the same type.

The calculus Λ of $\beta\eta$ equality is axiomatized with the usual axioms

$$(\beta) \qquad (\lambda_x a)b = a^x_b, \qquad \text{provided } b \text{ is free for } x \text{ in } a,$$

$$(\eta) \qquad \lambda_x a x = a, \qquad \text{provided } x \text{ is not free in } a,$$

and the axioms and rules for equality, i.e. $a = a$ and the rule of replacement of equals. It is not usually noted that the equality of α conversion can be proved from the remaining axioms as follows:

$$\begin{aligned} \lambda_x a &= \lambda_y (\lambda_x a) y, \text{ by } (\eta), \\ &= \lambda_y a^x_y, \text{ by } (\beta), \end{aligned}$$

where y is a variable not occurring in a.

4 Lambda terms for P-functionals

Let P be a finite ordinal. In what follows an interesting P will be greater than or equal to the ordinal 2. The set of *P-types* is defined inductively by specifying that P is a P-type and that if A and B are P-types, then $A \to B$, i.e. the set of all functions with domain A and codomain B, is a P-type. Symbols for P-types are types with a single atomic type P. It is clear that for P nonempty a P-type cannot be named by two different P-type symbols.

An element of a P-type is called a *P-functional*. It is clear that every P-functional is finite (i.e., its graph is a finite set of ordered pairs) and that in every P-type there are only finitely many P-functionals. For P-functionals we use the Greek letters φ, ψ, ..., φ_1, ...

Our aim is to define for every P-functional a closed term defining it, in a sense to be made precise. But before that we must introduce a series of preliminary definitions. In these definitions we take that the calculus Λ is built over types with a single atomic type, which we call p.

Let the type A_0 be p and let the type A_{n+1} be $A_n \to A_n$. For $i \geq 0$, let the type N_i be A_{i+2}, i.e. $(A_i \to A_i) \to (A_i \to A_i)$.

Let $x^0(y)$ be y and let $x^{n+1}(y)$ be $x(x^n(y))$. The terms $[n]_i$, called *Church numerals of type N_i*, are defined by

$$[n]_i =_{def} \lambda_{xy} x^n(y)$$

for $x : A_{i+1}$ and $y : A_i$.

For x, y and z all of type N_i, $u : A_{i+1}$, and v and w of type A_i, let

$$C_i =_{def} \lambda_{xyzuv} x(\lambda_w zuv)(yuv).$$

These are conditional function combinators, because in the calculus Λ one can prove

$$C_i[n]_i ab = \begin{cases} a & \text{if } n = 0 \\ b & \text{if } n \neq 0 \end{cases}$$

For $x : N_{i+1}$, y and z of type A_{i+1}, and u and v of type A_i, let

$$R_i =_{def} \lambda_{xy} x(\lambda_{zu} y(zu))(\lambda_v v).$$

These combinators reduce the types of numerals; namely, in Λ one can prove

$$R_i[n]_{i+1} = [n]_i.$$

For x and y of type N_{i+1}, let the exponentiation combinators be defined by

$$E_i =_{def} \lambda_{xy} x(R_i y).$$

In Λ one can prove

$$E_i[n]_{i+1}[m]_{i+1} = [m^n]_i.$$

For $E_i ab$ we use the abbreviation b^a.

For x and y of type N_i, $z : A_{i+1}$ and $u : A_i$, let the addition and multiplication combinators be defined by

$$\begin{aligned} S_i &=_{def} \lambda_{xyzu} xz(yzu), \\ M_i &=_{def} \lambda_{xyzu} x(yz)u. \end{aligned}$$

In Λ one can prove

$$\begin{aligned} S_i[n]_i[m]_i &= [n+m]_i, \\ M_i[n]_i[m]_i &= [n \cdot m]_i. \end{aligned}$$

For $M_i ab$ we use the abbreviation $a{\cdot}b$.

For x, y and z of type N_i, and $u : N_{i+1}$, let the pairing and projection combinators be defined by

$$\begin{aligned} \Pi_i &=_{def} \lambda_{xyz} C_i zxy, \\ \pi_i^1 &=_{def} \lambda_u u[0]_i, \\ \pi_i^2 &=_{def} \lambda_u u[1]_i. \end{aligned}$$

In Λ one can prove

$$\begin{aligned} \pi_i^1(\Pi_i ab) = a, \\ \pi_i^2(\Pi_i ab) = b. \end{aligned}$$

For $x : N_{i+1}$ and $y : N_{i+3}$, let

$$\begin{aligned} T_i &=_{def} \lambda_x \Pi_i(S_i[1]_i(\pi_i^1 x))(\pi_i^1 x), \\ H_i &=_{def} \lambda_y y T_i(\Pi_i[0]_i[0]_i), \\ P_i &=_{def} \lambda_y \pi_i^2(H_i y). \end{aligned}$$

The terms T_i and H_i are auxiliary, while the terms P_i are predecessor combinators, because, for $n \geq 1$, one can prove in Λ

$$\begin{aligned} P_i[n]_{i+3} &= [n-1]_i, \\ P_i[0]_{i+3} &= [0]_i. \end{aligned}$$

Typed terms corresponding to all the terms C_i, R_i, up to P_i, may be found in [2] (cf. [7]).

For x and y of type N_i, $z : A_{i+1}$, and u and v of type A_i, let

$$Z_{i+1} =_{def} \lambda_{xyzu} x(\lambda_v yzu)(zu).$$

These combinators raise the types of numerals for 0 and 1; namely, in Λ one can prove

$$Z_{i+1}[0]_i = [0]_{i+1},$$
$$Z_{i+1}[1]_i = [1]_{i+1}.$$

The equality (η) is essential to prove this.

For $x : N_i$, let

$$D_i^0 =_{def} \lambda_x C_i x[0]_i[1]_i$$

and for $k \geq 1$ and $i \geq 3k$ let

$$D_i^k =_{def} \lambda_x C_i x[1]_i Z_i(Z_{i-1}(Z_{i-2}(D_{i-3}^{k-1}(P_{i-3}x)))).$$

These combinators check whether a numeral stands for k; namely, for $n \geq 0$, one can prove in Λ

$$D_i^k[n]_i = \begin{cases} [0]_i & \text{if } n = k \\ [1]_i & \text{if } n \neq k. \end{cases}$$

For every P-type symbol A, let A^i be the type obtained from A by substituting N_i for P. Now we are ready to define for every P-functional $\varphi \in A$ a closed term $\varphi^\lambda : A^i$.

Take a P-functional $\varphi \in A$, where A is $B_1 \to (\ldots \to (B_k \to P)\ldots)$. By induction on the complexity of the P-type symbol A we define a natural number $\kappa(\varphi)$ and for every $i \geq \kappa(\varphi)$ we define a term $\varphi^\lambda : A^i$.

If A is P, then φ is an ordinal n in P. Then $\kappa(n) = 0$ and $n^\lambda : N_i$ is $[n]_i$ for every $i \geq 0$.

Suppose $k \geq 1$ and B_1 is $B \to (C \to P)$. It is enough to consider this case, which gives the gist of the proof. When B_1 is $C_1 \to (C_2 \to \ldots (C_l \to P)\ldots)$ for l different from 2 we proceed analogously, but with more notational complications if $l \geq 3$. For $B = \{\beta_1, \ldots, \beta_m\}$ and $C = \{\gamma_1, \ldots, \gamma_r\}$, by the induction hypotheses, we have defined $\kappa(\beta_1)$, ..., $\kappa(\beta_m)$, $\kappa(\gamma_1)$, ..., $\kappa(\gamma_r)$, for every $i \geq \kappa(\beta_1)$ we have defined β_1^λ, and analogously for β_2, ..., β_m, γ_1, ..., γ_r. For $B_1 = \{\psi_1, \ldots, \psi_q\}$, let $\varphi(\psi_j) = \xi_j \in B_2 \to (\ldots \to (B_k \to P)\ldots)$. (Note that φ is not necessarily one-one.) By the induction hypothesis, we have defined $\kappa(\xi_1), \ldots, \kappa(\xi_q)$, for every $i \geq \kappa(\xi_1)$ we have defined ξ_1^λ, and analogously for $\xi_2, \ldots, \xi_q$.

Let now

$$\begin{array}{llll}
(\psi_1(\beta_1))(\gamma_1) = d_1 \in P, & (\psi_1(\beta_2))(\gamma_1) = d_{r+1} \in P, & \ldots & (\psi_1(\beta_m))(\gamma_1) = d_{(m-1)r+1} \in P \\
(\psi_1(\beta_1))(\gamma_2) = d_2 \in P, & (\psi_1(\beta_2))(\gamma_2) = d_{r+2} \in P, & \ldots & (\psi_1(\beta_m))(\gamma_2) = d_{(m-1)r+2} \in P \\
\vdots & \vdots & & \vdots \\
(\psi_1(\beta_1))(\gamma_r) = d_r \in P, & (\psi_1(\beta_2))(\gamma_r) = d_{2r} \in P, & \ldots & (\psi_1(\beta_m))(\gamma_r) = d_{mr} \in P
\end{array}$$

Let $n_1 = 2^{d_1} \cdot 3^{d_2} \cdot \ldots \cdot p_{mr}^{d_{mr}}$, where p_{mr} is the mr-th prime number. Analogously, we obtain the natural numbers $n_2, \ldots, n_q$, all different, that correspond to $\psi_2, \ldots, \psi_q$.

We can now define $\kappa(\varphi)$ as

$$max\{3 \cdot max\{n_1, \ldots, n_q\}+1, \kappa(\beta_1), \ldots, \kappa(\beta_m), \kappa(\gamma_1), \ldots, \kappa(\gamma_r), \kappa(\xi_1), \ldots, \kappa(\xi_q)\}.$$

For every $i \geq \kappa(\varphi)$ and for $x_1 : B_1^i$, let the term t be defined as

$$[2]_i^{x_1 \beta_1^\lambda \gamma_1^\lambda} \cdot [3]_i^{x_1 \beta_1^\lambda \gamma_2^\lambda} \cdot \ldots \cdot [p_{mr}]_i^{x_1 \beta_m^\lambda \gamma_r^\lambda} : N_{i-1}.$$

For $x_2 : B_2^i, \ldots, x_k : B_k^i$, let

$$\begin{aligned}
Q_1 &=_{def} C_i(Z_i(D_{i-1}^{n_1} t))(\xi_1^\lambda x_2 \ldots x_k) Q_2, \\
Q_2 &=_{def} C_i(Z_i(D_{i-1}^{n_2} t))(\xi_2^\lambda x_2 \ldots x_k) Q_3, \\
&\vdots \\
Q_{q-1} &=_{def} C_i(Z_i(D_{i-1}^{n_{q-1}} t))(\xi_{q-1}^\lambda x_2 \ldots x_k)(\xi_q^\lambda x_2 \ldots x_k).
\end{aligned}$$

We can now, finally, define φ^λ as $\lambda_{x_1 \ldots x_k} Q_1$.

Next we define by induction on the complexity of the P-type symbol A, when a P-functional $\varphi \in A$ is *i-defined* by a term $a : A^i$.

We say that a closed term $a : N_i$ *i-defines* an ordinal $n \in P$ iff in Λ we can prove $a = [n]_i$.

For a P-functional $\varphi \in B \to C$ we say that $a : B^i \to C^i$ *i-defines* φ iff, for every $\psi \in B$ and every $b : B^i$, if b i-defines ψ, then $ab : C^i$ i-defines $\varphi(\psi) \in C$.

We can now prove the following lemma.

Lemma 4.1 *For every $i \geq \kappa(\varphi)$, the P-functional $\varphi \in A$ is i-defined by $\varphi^\lambda : A^i$.*

Proof: We proceed by induction on the complexity of the P-type symbol A. The case when A is P is trivial.

Let now A be of the form $B_1 \to (\ldots \to (B_k \to P) \ldots)$ for $k \geq 1$, let $B_1 = \{\psi_1, \ldots, \psi_q\}$, and let everything else be as in the inductive step of the definition of φ^λ. Suppose $b_1 : B_1^i$ i-defines ψ_1. We have to check that $\varphi^\lambda b_1$ i-defines $\varphi(\psi_1) = \xi_1$.

By the induction hypothesis we have that $\beta_1^\lambda, \ldots, \beta_m^\lambda, \gamma_1^\lambda, \ldots, \gamma_r^\lambda, \xi_1^\lambda, \ldots, \xi_q^\lambda$ i-define $\beta_1, \ldots, \beta_m, \gamma_1, \ldots, \gamma_r, \xi_1, \ldots, \xi_q$, respectively. Then we have

$$\begin{aligned}
\varphi^\lambda b_1 &= (\lambda_{x_1 \ldots x_k} C_i(Z_i(D_{i-1}^{n_1} t))(\xi_1^\lambda x_2 \ldots x_k) Q_2) b_1 \\
&= \lambda_{x_2 \ldots x_k} C_i(Z_i(D_{i-1}^{n_1} t_{b_1}^{x_1}))(\xi_1^\lambda x_2 \ldots x_k)(Q_2)_{b_1}^{x_1}.
\end{aligned}$$

For the closed term $t^{x_1}_{b_1}$ we have

$$t^{x_1}_{b_1} = [2]_i^{b_1\beta_1^\lambda\gamma_1^\lambda} \cdot [3]_i^{b_1\beta_1^\lambda\gamma_2^\lambda} \cdot \ldots \cdot [p_{mr}]_i^{b_1\beta_m^\lambda\gamma_r^\lambda}.$$

It follows by the induction hypothesis that $b_1\beta_1^\lambda\gamma_1^\lambda$ i-defines d_1, which means that in Λ we can prove $b_1\beta_1^\lambda\gamma_1^\lambda = [d_1]_i$. We proceed analogously with the other exponents. So in Λ we can prove $t^{x_1}_{b_1} = [n_1]_{i-1}$. Hence in Λ we have $D^{n_1}_{i-1}t^{x_1}_{b_1} = [0]_{i-1}$, and we conclude that

$$\begin{array}{rcl} \varphi^\lambda b_1 & = & \lambda_{x_2\ldots x_k}\xi_1^\lambda x_2\ldots x_k \\ & = & \xi_1^\lambda,\ \text{by } (\eta). \end{array}$$

So $\varphi^\lambda b_1$ i-defines ξ_1.

Suppose now $b_2 : B^i_1$ i-defines ψ_2. Then in Λ we have

$$\varphi^\lambda b_2 = \lambda_{x_2\ldots x_k}C_i(Z_i(D^{n_1}_{i-1}t^{x_1}_{b_2}))(\xi_1^\lambda x_2\ldots x_k)(C_i(Z_i(D^{n_2}_{i-1}t^{x_1}_{b_2}))(\xi_2^\lambda x_2\ldots x_k)(Q_3)^{x_1}_{b_2}).$$

Since in Λ we can prove $t^{x_1}_{b_2} = [n_2]_{i-1}$, we can also prove $D^{n_1}_{i-1}t^{x_1}_{b_2} = [1]_{i-1}$, and we conclude that

$$\varphi^\lambda b_2 = \lambda_{x_2\ldots x_k}C_i(Z_i(D^{n_2}_{i-1}[n_2]_{i-1}))(\xi_2^\lambda x_2\ldots x_k)(Q_3)^{x_1}_{b_2}.$$

Finally, we obtain as above that $\varphi^\lambda b_2$ i-defines ξ_2. We proceed analogously for $\psi_3, \ldots, \psi_q$. □

This lemma does not mean that we can i-define all P-functionals simultaneously for some i. But we can always find such an i for finitely many P-functionals.

5 P-models

A model based on $P = \{0, \ldots, h-1\}$, with $h \geq 2$, for the calculus Λ built over types with a single atomic type p will be defined as in [5].

An *assignment* is a function f assigning to a variable $x : A$ of Λ a functional $f(x)$ in the P-type A^p_P, where A^p_P is obtained from A by substituting P for p. For an assignment f and a variable y, the assignment f^y_α is defined by

$$f^y_\alpha(x) = \begin{cases} \alpha & \text{if } x \text{ is } y \\ f(x) & \text{if } x \text{ is not } y. \end{cases}$$

If F is the set of all P-functionals, then the *P-model* is a pair $\langle F, V\rangle$ such that V maps the pairs (a, f), with a a term and f an assignment, into F. We write $V_{a,f}$ instead of $V(a,f)$. The function V must satisfy the conditions

$$
\begin{aligned}
&V_{x,f} = f(x), \\
&V_{ab,f} = V_{a,f}(V_{b,f}), \\
\text{for } x : A \text{ and } \alpha : A_P^p, \quad &V_{\lambda_x a,f}(\alpha) = V_{a,f_\alpha^x}.
\end{aligned}
$$

There is exactly one such function V.

Let $a : A$ be a term such that $x_1 : A_1, \ldots, x_n : A_n$ are all the variables, both free and bound, occurring in a. Let f be an assignment, and for every $j \in \{1, \ldots, n\}$ let b_j i-define $f(x_j)$. Finally, let $\underline{a}$ be the type-instance of a obtained by substituting N_i for p. The type of $\underline{a}$ is $(A_P^p)^i$. Then we can prove the following lemma.

Lemma 5.1 *The term* $\underline{a}\frac{x_1 \ldots x_n}{b_1 \ldots b_n}$ *i-defines* $V_{a,f}$.

The proof proceeds by a straightforward induction on the complexity of the term a.

Of course, when a is closed, $V_{a,f}$ does not depend on f, and has the same value for all assignments f. So, for closed terms a, we can write V_a instead of $V_{a,f}$, and we shall do so from now on.

As an immediate corollary of Lemma 5.1 we obtain the following lemma.

Lemma 5.2 *If a is closed, then $\underline{a}$ i-defines V_a.*

6 Böhm's Theorem with types

We are now ready to prove our analogue of Böhm's Theorem for the typed lambda calculus Λ, which is not necessarily built over types with a single atomic type.

Theorem 6.1 *If a and b are of the same type and $a = b$ is not provable in Λ, then for every two terms c and d of the same type one can construct type-instances a' and b' of a and b, respectively, and terms $h_1, \ldots, h_n$, $n \geq 0$, and also find variables $x_1, \ldots, x_m$, $m \geq 0$, such that*

$$(\lambda_{x_1 \ldots x_m} a')h_1 \ldots h_n = c,$$

$$(\lambda_{x_1 \ldots x_m} b')h_1 \ldots h_n = d$$

are provable in Λ.

Proof: Let a_1 and b_1 be type-instances of a and b, respectively, obtained by substituting p for all atomic types. It is easy to see that $a = b$ is provable in Λ iff $a_1 = b_1$ is provable in Λ.

Let $x_1, \ldots, x_m$ be all the free variables in a_1 or b_1. Then since $a_1 = b_1$ is not provable in Λ, the equality $\lambda_{x_1 \ldots x_m} a_1 = \lambda_{x_1 \ldots x_m} b_1$ is not provable in Λ. Let a_2 be $\lambda_{x_1 \ldots x_m} a_1$ and let b_2 be $\lambda_{x_1 \ldots x_m} b_1$.

It follows from a theorem of [9] (Theorem 2, p. 187) and [10] (Theorem 2, p. 21) that if $a_2 = b_2$ is not provable in Λ, then there exists a P-model $\langle F, V \rangle$ such that $V_{a_2} \neq V_{b_2}$. Soloviev's and Statman's theorem doesn't mention exactly P-models, which are based on the full type structure built over an ordinal P, but instead it mentions completely analogous models based on the full type structure built over a finite set S.

We can always name the elements of S by ordinals so that S becomes an ordinal P. Moreover, for every two distinct elements s_1 and s_2 of S we can always name the elements of S so that s_1 is named by 0 and s_2 is named by 1. This means that the elements of S can always be named by elements of P so that in the P-model $\langle F, V \rangle$ above there are P-functionals $\varphi_1, \ldots, \varphi_k$, $k \geq 0$, such that

$$((V_{a_2}(\varphi_1))(\varphi_2)) \ldots (\varphi_k) = 0,$$

$$((V_{b_2}(\varphi_1))(\varphi_2)) \ldots (\varphi_k) = 1.$$

Take an even $i \geq max\{\kappa(\varphi_1), \ldots, \kappa(\varphi_k)\}$. By Lemma 4.1, the closed terms $\varphi_1^\lambda, \ldots, \varphi_k^\lambda$ i-define $\varphi_1, \ldots, \varphi_k$, respectively. By Lemma 5.2, the term $\underline{a}_2$ i-defines V_{a_2} and $\underline{b}_2$ i-defines V_{b_2}. It follows that in Λ we can prove $\underline{a}_2 \varphi_1^\lambda \ldots \varphi_k^\lambda = [0]_i$ and $\underline{b}_2 \varphi_1^\lambda \ldots \varphi_k^\lambda = [1]_i$.

For $x : A_i$, $y : A_{i-1}$ and $z : A_{i-2}$ we can prove in Λ

$$[0]_i(\lambda_{xyz} yz)(\lambda_{yz} z) = [0]_{i-2},$$

$$[1]_i(\lambda_{xyz} yz)(\lambda_{yz} z) = [1]_{i-2}.$$

So there are closed terms $c_1, \ldots, c_i$ such that in Λ we can prove

$$\underline{a}_2 \varphi_1^\lambda \ldots \varphi_k^\lambda c_1 \ldots c_i = [0]_0,$$

$$\underline{b}_2 \varphi_1^\lambda \ldots \varphi_k^\lambda c_1 \ldots c_i = [1]_0.$$

Let the left-hand sides of these two equalities be a_3 and b_3, respectively.

Take now c and d of type A and take the type-instances a_4 and b_4 of a_3 and b_3, respectively, obtained by substituting A for p. For $u : A$ we can prove in Λ

$$a_4(\lambda_u d)c = c,$$

$$b_4(\lambda_u d)c = d.$$

The terms a_4 and b_4 are of the form $(\lambda_{x_1 \ldots x_n} a')h_1 \ldots h_{k+i}$ and $(\lambda_{x_1 \ldots x_n} b')h_1 \ldots h_{k+i}$. If $(N_i)_A^p$ is obtained by substituting A for p in N_i, then a' is a type-instance of a obtained by substituting $(N_i)_A^p$ for every atomic type. $\square$

Since the procedure for constructing $h_1, \ldots, h_n$ in the proof of Theorem 6.1 can be pretty involved, it may be useful to illustrate this procedure with an example. For this example we take two terms unequal in Λ that we mentioned in Section 2 (this is the more involved of the examples given there).

Example: Let a and b be $\lambda_x x\lambda_y(x\lambda_z y)$ and $\lambda_x x\lambda_y(x\lambda_z z)$, respectively, with $x : (p \to p) \to p$, $y : p$ and $z : p$. Since all the atomic types of a and b are already p, and since these two terms are closed, we have that a_2 is a and b_2 is b.

The P-model falsifying $a = b$ has $P = \{0, 1\}$ and $P \to P = \{\psi_1, \psi_2, \psi_3, \psi_4\}$, where

$$\begin{array}{l} \psi_1(0) = \psi_1(1) = 0, \\ \psi_2(0) = \psi_2(1) = 1, \\ \psi_3(0) = 0, \quad \psi_3(1) = 1, \\ \psi_4(0) = 1, \quad \psi_4(1) = 0. \end{array}$$

For $\varphi \in (P \to P) \to P$ defined by

$$\varphi(\psi_1) = 1, \quad \varphi(\psi_2) = \varphi(\psi_3) = \varphi(\psi_4) = 0$$

we have $V_a(\varphi) = 0$ and $V_b(\varphi) = 1$.

Then

$$\begin{array}{ll} n_1 = 2^0 \cdot 3^0 = 1 & \text{corresponds to } \psi_1, \\ n_2 = 2^1 \cdot 3^1 = 6 & \text{corresponds to } \psi_2, \\ n_3 = 2^0 \cdot 3^1 = 3 & \text{corresponds to } \psi_3, \\ n_4 = 2^1 \cdot 3^0 = 2 & \text{corresponds to } \psi_4, \end{array}$$

and $\kappa(\varphi) = 19$. For every $i \geq 19$ and for $x_1 : N_i \to N_i$, the term t is defined as $[2]_i^{x_1[0]_i} \cdot [3]_i^{x_1[1]_i} : N_{i-1}$. The term φ^λ is defined as

$$\lambda_{x_1} C_i(Z_i(D^1_{i-1}t))[1]_i(C_i(Z_i(D^6_{i-1}t))[0]_i(C_i(Z_i(D^3_{i-1}t))[0]_i[0]_i)).$$

The terms $\underline{a}$ and $\underline{b}$ are like a and b with $x : (N_{20} \to N_{20}) \to N_{20}$, $y : N_{20}$ and $z : N_{20}$, and let i in φ^λ be 20. Then in Λ we can prove $\underline{a}\varphi^\lambda = [0]_{20}$ and $\underline{b}\varphi^\lambda = [1]_{20}$. The remaining steps in the construction of a_3 and b_3 are straightforward, and we shall not pursue this example further.

By taking that for x and y of the same type the term c is $\lambda_{xy}x$ and d is $\lambda_{xy}y$, we obtain the following refinement of Theorem 6.1.

Theorem 6.2 *If a and b are of the same type and $a = b$ is not provable in Λ, then for every two terms e and f of the same type one can construct type-instances a' and b' of a and b, respectively, and closed terms $h_1, \ldots, h_l$, $l \geq 0$, and also find variables $x_1, \ldots, x_m$, $m \geq 0$, such that*

$$(\lambda_{x_1 \ldots x_m} a')h_1 \ldots h_l ef = e,$$

$$(\lambda_{x_1 \ldots x_m} b')h_1 \ldots h_l ef = f$$

are provable in Λ.

It is clear that if a and b are closed, we need not mention in this theorem the variables $x_1, \dots, x_m$ and we can omit the λ-abstraction $\lambda_{x_1 \dots x_m}$.

Although our proof of Theorem 6.1 relies on the equality (η) at some key steps (as we noted in connection with the combinator Z_{i+1}), it is possible to derive a strengthening of this theorem, as well as of Theorem 6.2, where Λ is replaced by Λ_β, which is Λ minus (η) and plus the equality of α conversion. We learned how to obtain this strengthening from Alex Simpson.

First note that if a term a is in both contracted and expanded $\beta\eta$ normal form, and $a = b$ in Λ, then $a = b$ in Λ_β. For if $a = b$ in Λ, then, since a is in contracted $\beta\eta$ normal form, there is a term a' such that b β-reduces to a' and a' η-reduces by contractions to a. But then, since a is also in expanded $\beta\eta$ normal form, a' must be the same term as a. So $a = b$ in Λ_β.

Then, as we did to derive Theorem 6.2, take in Theorem 6.1 that c is $\lambda_{xy}x$ and d is $\lambda_{xy}y$ for x and y of atomic type p. The terms c and d are then in both contracted and expanded $\beta\eta$ normal form, and hence it is easy to infer Simpson's strengthening mentioned above by instantiating p with an arbitrary type.

To formulate below a corollary of Theorem 6.1 we must explain what it means to extend Λ with a new axiom. Let a and b be of type A, and let a' and b' be type-instances of a and b respectively. Then assuming $a = b$ as a new axiom in Λ means assuming also $a' = b'$. In other words, $a = b$ is assumed as an axiom schema, atomic types being understood as schematic letters. The postulate (β) and (η) are also assumed as axiom schemata, in the same sense. We could as well add to Λ a new rule of substitution for atomic types. The calculus Λ is closed under this substitution rule (i.e., this rule is admissible, though not derivable from the other rules). And any extension of Λ we envisage should be closed under this rule. The rule of substitution of types says that atomic types are variables.

We can now state the following corollary of Theorem 6.1.

Corollary *If* $a = b$ *is not provable in* Λ*, then in* Λ *extended with* $a = b$ *we can prove every formula* $c = d$.

Acknowledgement. We would like to thank Alex Simpson for reading a previous version of this paper, and for making a very helpful suggestion (noted in Section 6). We are also grateful to Slobodan Vujošević for his careful reading of the text and for his comments.

References

[1] H.P. Barendregt, *The Lambda Calculus: Its Syntax and Semantics*, North-Holland, Amsterdam, 1981, revised edition 1984.

[2] E. Barendsen, *Representation of Logic, Data Types and Recursive Functions in Typed Lambda Calculi*, Doctoraal Scripte, Faculteit Wiskunde en Informatica, Katholicke Universiteit Nijmegen, 1989.

[3] C. Böhm, *Alcune proprietà delle forme β-η-normali nel λ-K-calcolo*, Pubblicazioni dell'Istituto per le Applicazioni del Calcolo, Rome, **696** (1968), 19 pp.

[4] H.B. Curry, J.R. Hindley and J.P. Seldin, *Combinatory Logic, Volume* II, North-Holland, Amsterdam, 1972.

[5] H. Friedman, *Equality between functionals*, in: R. Parikh ed., *Logic Colloquium '73*, Lecture Notes in Math. **453**, Springer, Berlin, 1975, 22-37.

[6] J.-L. Krivine, *Lambda-calcul: Types et modèles*, Masson, Paris, 1990 (English translation, Ellis Horwood, 1993).

[7] H. Schwichtenberg, *Definierbare Funktionen im Lambda-Kalkül mit Typen*, Arch. math. Logik Grundlagenforsch. **17** (1976), 113-114. (We know this paper only from references.)

[8] A.K. Simpson, *Categorical completeness results for the simply-typed lambda-calculus*, in: M. Dezani-Ciancaglini and G. Plotkin eds, *Typed Lambda Calculi and Applications (Edinburgh, 1995)*, Lecture Notes in Comput. Sci. **902**, Springer, Berlin, 1995, 414-427.

[9] S.V. Soloviev, *The category of finite sets and cartesian closed categories* (in Russian), Zapiski nauchn. sem. LOMI **105** (1981), 174-194 (English translation in J. Soviet Math. **22**, 1983, 1387-1400).

[10] R. Statman, *Completeness, invariance and λ-definability*, J. Symbolic Logic **47** (1982), 17-26.

[11] R. Statman, *λ-definable functionals and $\beta\eta$-conversion*, Arch. math. Logik Grundlagenforsch. **23** (1983), 21-26.