Counting a type's inhabitants

Thiago Mendonça & Washington Ribeiro

July 8, 2015

Contents

1 Introduction

Surprisingly, the Intuitionist Implicational Logic has decidability. One time one gives a proposition in this logic is possible to decide, in a finite number of steps, if it holds or not. Which is equivalent, by the Curry-Howard isomorphism, to find a closed term M that has a type τ (by TA_λ deductions).

So the type's inhabitants problem became a important issue, on the view that when we built an algorithm to decide (and count) the inhabitants of a given type we are indeed showing the decidibilty of the Intuitionist Implicational Logic. And it will be useful to on the task of obtain negative answers, for example to prove that the "Peirce's Law" isn't a theorem of this logic.

But if the question is changed to how many terms in **normal form** is inhabitants of a given type, there are important properties and a well known algorithm from *Ben-Yelles (1979)* that solves the problem. This document will present this algorithm in a informal view in the first section and the second and third will show its soundness and completeness.

All content of this monography is entirely based on the chapter 8 of the book "Basic Simple Type Theory (1997)" of *J. Roger Hindley*. The lemmas that are inside the book but out of the mentioned chapter will be cited by the label their appear on the book (ex. **2B2**).

2 Inhabitants

Notation 1. *The following notations are convinient:*

(i) $\boldsymbol{Habs}_u(\tau)$ *denotes the set of all untyped inhabitants of* τ *and* $\boldsymbol{Habs}_t(\tau)$ *denotes the set of all typed ones,* [1]

(ii) $\boldsymbol{Nhabs}_u(\tau)$ *e* $\boldsymbol{Nhabs}_t(\tau)$ *characterises respectively the set of untyped e typed inhabitants in a* β*-normal form. They are designed by* β***-normal inhabitants***, [2]

(iii) the set of all (typed and untyped) $\beta\eta$***-normal inhabitants*** *is denotated by* $\boldsymbol{Nhabs}_\eta(\tau)$.

[1] If there is no confusion, $\mathbf{Habs}(\tau)$ can be used for both.
[2] If there is no confusion, $\mathbf{Nhabs}(\tau)$ can be used for both.

Lemma 1. *If $M^\tau \in Nhabs_t(\tau)$ then $M^{\not\tau} \in Nhabs_u(\tau)$; further, type-erasing mapping is a one-to-one correspondence between the typed and the untyped β-normal inhabitants of τ (moudulo $\equiv_\alpha$). The same holds for $\beta\eta$-normal inhabitants.*

Proof. Let $M \in \beta$-nf or $M \in \beta\eta$-nf; then M inhabits τ iff there exists a proof Δ of $\mapsto M : \tau$, and by the lemma **2B3** this Δ is uniquely determined by M. And such proofs correpond one-to-one with typed closed term by the theorem **5A7**. $\square$

Definition 1 (Cardinality of $\mathbb{S}$)**.** *The number (0,1,2,... or ∞) of members of a set $\mathbb{S}$, counted modulo $\equiv_\alpha$ if $\mathbb{S}$ is a set of λ-terms, is called the **cardinality** of $\mathbb{S}$ which is denoted by $\#(\mathbb{S})$. For $\#(Nhabs(\tau))$ and $\#(Nhabs_\eta(\tau))$ it will be denoted respectively by $\#(\tau)$ and $\#_\eta(\tau)$.*

Definition 2 (Counting, enumerating)**.** *A distinction will be made between **couting** and **enumerating** a set $\mathbb{S}$ of β-normal inhabitants of a type τ:*

*(i) to **count** $\mathbb{S}$ will mean to compute $\#(\mathbb{S})$ after a finite number of steps (even when $\#(\mathbb{S}) = \infty$);*

*(ii) to **enumerate** or **list** $\mathbb{S}$ will mean enumerate $\mathbb{S}$ in the usual recursion-theoretic sense, i.e to output a sequence consisting of all the members of $\mathbb{S}$ (and non-member to), continuing for ever if $\mathbb{S}$ is infinite.*

Lemma 2 (Structure of a typed β-nf)**.** *Let Γ be a type context. Every β-nf $N^\alpha \in \mathbb{TT}(\Gamma)$ can be expressed uniquely in the form:*

(i) $(\lambda x_1^{\tau_1} ... x_m^{\tau_m} \,.\, (v^{(\rho_1 \to \cdots \rho_n \to \tau^*)} M_1^{\rho_1} \cdots M_n^{\rho_n})^{\tau^*})^{(\tau_1 \to \cdots \tau_m \to \tau^*)}$, where $m \geq 0$, $n \geq 0$, and

(ii) $\tau \equiv \tau_1 \to \cdots \tau_m \to \tau^*$ for some τ^*, possibly composite, and

(iii) each $M_j^{\rho_j}$ is a β-nf that is typed relative to $\Gamma \cup \{x_1 : \tau_1, \cdots, x_m : \tau_m\}$

Proof. Induction in $N^{\not\alpha}$:

(IB) : $N^\alpha \equiv x^\alpha$, x is in β-nf and $x \in \mathbb{TT}(\Gamma)$

(Case 1) : $N^{\not\alpha} \equiv AB$ A is not an abstraction because if it was, AB would not be in β-nf.

Without loss of generality, $A \equiv vH_1 \cdots H_k$.

By hipothesis of induction, $A^{\gamma\to\alpha} \in \mathbb{TT}(\Gamma_1)$ and $B^\gamma \in \mathbb{TT}(\Gamma_2)$. Using α equivalence to avoid clashes, $N^\alpha \in \mathbb{TT}(\Gamma_1 \cup \Gamma_2)$ is in the form $(v^{\rho_1\to\cdots\rho_k\to\gamma\to\alpha} H_1^{\rho_1} \cdots H_k^{\rho_k} B^\gamma)^\alpha$

(Case 2) : $N^{\not\alpha} \equiv \lambda x \,.\, A$

By hipothesis of induction, $A^\gamma \in \mathbb{TT}(\Gamma_1)$ and it has the anounced form.

Thus, $(\lambda x^\omega \,.\, A^\gamma)^{\omega\to\gamma} \in \mathbb{TT}(\Gamma_1 - \{x : \omega\})$ has the same form.

$\square$

Definition 3 (Long β-nf's)**.** *A typed β-nf M^τ is called **long** or **maximal** iff every variable-occurence $\underline{z} \in M^\tau$ is followed by the longest sequence of arguments allowed by its type, i.e. iff each component with form $(zP_1...P_n)(n \geq 0)$ that is not in a function position has atomic type. An untyped β-nf M is called **long** relative to a type τ iff it is a type-erasure of a typed long β-nf M^τ (by the lemma 1 M^τ is unique).*

Notation 2. *The sets of all long normal inhabitants of τ (typed or untyped) will both be called:*

$$Long(\tau).$$

Example 1. *For the type $\tau \equiv ((a \to b) \to c) \to (a \to b) \to c$:*

(i) $M^\tau \equiv \lambda x^{(a\to b)\to c} y^{a\to b} \cdot x^{(a\to b)\to c} y^{a\to b}$ isn't in a long normal form because y occurs in a non function position and has a composite type;

(ii) $N^\tau \equiv \lambda x^{(a\to b)\to c} y^{a\to b} \cdot x^{(a\to b)\to c} (\lambda z \cdot y^{a\to b} z^a)$ is in a long normal form.

Lemma 3 (Completeness of Long(τ))**.** *Every normal inhabitant of τ can be η-expanded to a long normal inhabitant of τ. And this long inhabitant is unique (modulo $\equiv_\alpha$); ie*

$$\{M^\tau, N^\tau \in Long(\tau) \ \text{and}\ M^\tau =_\eta N^\tau\} \Longrightarrow M^\tau \equiv_\alpha N^\tau$$

In particular

$$Nhabs(\tau) = \emptyset \Longleftrightarrow Long(\tau) = \emptyset.$$

Proof. Let $P^\tau \in Nhabs(\tau)$ so η-expand P^τ to $P^{\tau+} \in Long(\tau)$. We must show that $P^{\tau+}$ is unique

$$M^\tau \in Long(\tau),\ \ M^\tau \twoheadrightarrow_\eta P^\tau \quad \Longrightarrow \quad M^\tau \equiv_\alpha P^{\tau+}$$

Suppose P^τ contains a component $(y\, Q_1 \cdots Q_n)^\sigma$ that is not long, it is not in function position and $\sigma \equiv \sigma_1 \to \cdots \to \sigma_k \to a \ \ (k \geq 1)$. Given new variables $z_1, ..., z_k$ not occuring in $P^{\cancel{\tau}}$, replace this component by

$$(\lambda z_1^{\sigma_1} \ldots z_k^{\sigma_k} \cdot ((y\, Q_1 \cdots Q_n)^\sigma\, z_1^{\sigma_1} \cdots z_k^{\sigma_k})^a)^\sigma$$

And make similar replacements until there are no short components in P^τ. Call the result $P^{\tau+}$. Each replacement may introduce new short components with types $\sigma_1, ..., \sigma_k$, but these types are shorter than σ so the replacement process will terminate. Then:

- this replacements are η-expansions and its result is still a β-nf
  in $(\lambda z_1^{\sigma_1} \ldots z_k^{\sigma_k} \cdot ((y\, Q_1 \cdots Q_n)^\sigma\, z_1^{\sigma_1} \cdots z_k^{\sigma_k})^a)^\sigma$ the β-reductions could ocurr just inside $Q_1 \cdots Q_n$, but it is contraditory with the fact that $(y\, Q_1 \cdots Q_n)^\sigma$ is β-nf;
- the proof is complete with rotine induction in $|P^\tau \equiv \lambda x_1^{\tau_1} \ldots x_m^{\tau_m} . v\, M_1^{\rho_1} \cdots M_n^{\rho_n}|$:
  - By IH, for each j, $Long(\tau) \ni M_j^{\rho_j +} \twoheadrightarrow_\eta M_j^{\rho_j}$ and $M_j^{\rho_j +}$ is unique
  - $P^{\tau+} \equiv \lambda x_1^{\tau_1} \ldots x_m^{\tau_m}\, z_1^{\sigma_1} \ldots z_k^{\sigma_k} . v\, M_1^{\rho_1 +} \cdots M_n^{\rho_n +}\, z_1^{\sigma_1} \ldots z_k^{\sigma_k}$
  - $P^{\tau+} \in Long(\tau),\ \ P^{\tau+} \twoheadrightarrow_\eta P^\tau$
  - $M^\tau \in Long(\tau),\ \ M^\tau \twoheadrightarrow_\eta P^\tau \quad \Longrightarrow \quad M^\tau \equiv_\alpha P^{\tau+}$

□

So one can define η-families (Fig. 1) that will be disjoint sets (according the lemma 4).

Definition 4 (η-family)**.**

Notation 3. *The set of all terms from η-reduction from M^τ is*

$$\{M^\tau\}_\eta$$

Lemma 4. *(i) The η-families of long typed terms of type τ partition $Nhabs(\tau)$ into non-overlapping finite subsets, each η-family contain one long member and one $\beta\eta$-nf.*

(ii) $\#(\tau)$ is infinte, finite or zero according to $\#_\eta(\tau)$ is infinite, finite or zero.

(iii) $\#_\eta(\tau) = \#(Long(\tau))$

Proof. The following itens summarize the demostration:

- $M^\tau \in \mathbb{TT}(\Gamma)$, $\{M^\tau\}_\eta$ is finite (by the lemma $\mathbf{CR}_\eta$);
- $\{M^\tau\}_\eta \subseteq \mathbb{TT}(\Gamma)$ (lemma **5B7.1**);

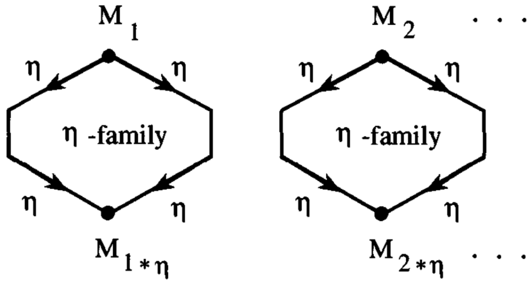


Figure 1: The disjoint η-families.

- if M^τ is a β-nf then so are all members of $\{M^\tau\}$ (typed analogue of the lemma **1C9.3**);

$$M^\tau \in Nhabs(\tau) \implies \{M^\tau\}_\eta \subseteq Nhabs(\tau);$$

- if M^τ is a β-nf its η-family contains exactly one $\beta\eta$-nf (typed analogue of the lemma **1C9.3**);
- each normal inhabitant of τ is in the η-family of exactly one long normal inhabitant (by the lemma 3);
- And by the **WN** lemma, $\#(\tau)$ is infinte, finite or zero according to $\#_\eta(\tau)$ is infinite, finite or zero.

$\square$

Definition 5 (Principal Inhabitants). *An untyped term M of type τ is called principal iff τ is the principal type of M. The inhabitant M^τ of τ is called principal iff the deduction of $\mapsto M\!\!/ : \tau$ is principal.*

The set of principal inhabitants of τ is

$$Princ(\tau)$$

And the set of principal inhabitants in β-nf of τ is

$$Nprinc(\tau)$$

Lemma 5. *M^τ is a principal β-nf inhabitant of τ iff M is an untyped β-nf inhabitant of τ.*

Proof. By the lemma **5B7.1(b)** M^τ is a β-nf iff $M\!\!/$ is a β-nf. And, by the theorem **5A7**, τ is a principal type of M iff M^τ has τ as principal type to. $\square$

Lemma 6. *Let $M^{\tau+}$ be the unique member of $Long(\tau)$ to which M^τ η-expands. Then*

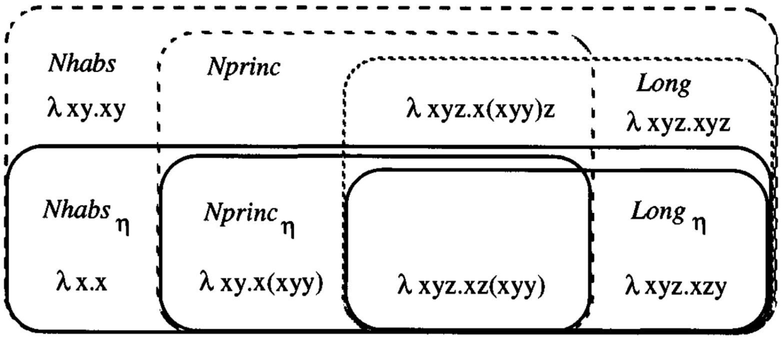
$$M^\tau \in Nprinc(\tau) \Longrightarrow M^{\tau+} \in Nprinc(\tau)$$

Proof. The η-expansion in the lemma 3 preserves principality so the types of $z_1, ..., z_k$ are determined by the type τ of the component that is replaced. $\square$

Now follow some examples of the relation between the classes of inhabitants:

$$\tau \equiv (a \to a \to a) \to a \to a \to a$$

$$\begin{array}{rlcl}
\text{(i)} & \lambda x^{a\to a\to a} \cdot x^{a\to a\to a} & \in & Nhabs_\eta - (Nprinc \cup Long) \\
\text{(ii)} & \lambda x^{a\to a\to a}\, y^a \cdot (x\, y)^{a\to a} & \in & Nhabs - Nhabs_\eta - (Nprinc \cup Long) \\
\text{(iii)} & \lambda x^{a\to a\to a}\, y^a\, z^a \cdot (x\, y\, z)^a & \in & Long - Nhabs_\eta - Nprinc \\
\text{(iv)} & \lambda x^{a\to a\to a}\, y^a \cdot (x\,(x\, y\, y))^{a\to a} & \in & Nhabs_\eta - Long_\eta \\
\text{(v)} & \lambda x^{a\to a\to a}\, y^a\, z^a \cdot (x\,(x\, y\, y)\, z)^a & \in & Nprinc \cap Long - Nhabs_\eta \\
\text{(vi)} & \lambda x^{a\to a\to a}\, y^a\, z^a \cdot (x\, z\,(x\, y\, y))^a & \in & Nprinc_\eta \cap Long_\eta \\
\text{(vii)} & \lambda x^{a\to a\to a}\, y^a\, z^a \cdot (x\, z\, y)^a & \in & Long_\eta - Nprinc_\eta
\end{array}$$



Follow the justifications:

(i) The term shown in (i) is clearly a $\beta\eta$-nf and is not long. It fails to encode a principal deduction for $\lambda x \cdot x$ because the **PT** $(\lambda x \cdot x)$ is not τ but $a \to a$.

(ii) This term is obtained by η-expanding the term in (i).

(iii) This term is obtained by η-expanding (i) until it becomes long.

(iv) This term is easily shown to be principal by the **PT algorithm** (**3E1**). It fails to be long because its second x from the right has only one argument.

(v) This term is obtained from (iv) by η-expansion; both occurrences of x now have two arguments.

(vi) This term is like (v) but z and xyy have been reversed to make it an η-nf.

(vii) This term is clearly long. However, its **PT** is not τ but

$$(a \to b \to c) \to b \to a \to c$$

Remark 1. *Some remarks are important:*

(i) $Habs(\tau) \neq \emptyset \iff Nhabs(\tau) \neq \emptyset$ *[**WN**]*

(ii) $Habs(\tau) \neq \emptyset \iff Princ(\tau) \neq \emptyset$ *[**converse PT**]*

(iii) $Habs(\tau) \neq \emptyset \;\;\not\Rightarrow\;\; Nprinc(\tau) \neq \emptyset$

(iv) $Princ(\tau) \neq \emptyset \;\;\not\Rightarrow\;\; Nprinc(\tau) \neq \emptyset$

The justifications for (iii) and (iv) will be given by the following note and example.

Note 1. *τ may have an inhabitant M, even a principal one, such that $PT(M)$ changes when M is reduced to $M*_\beta$.*

Example 2 ($\tau \equiv a \to a \to a$)**.** *We have:*

- *$Nhabs(\tau) = \{\lambda xy \cdot x, \lambda xy \cdot y\}$, but neither of these is principal;*
- *But there is a non-normal principal inhabitant: $(\lambda xyz \cdot \boldsymbol{K}(xy)(xz))\boldsymbol{I}$.*

3 Search strategies

The search algorithm will seek for normal long inhabitants of τ increasing a given parameter called **depth**. Follow some definitions, components and some informal examples of such algorithm in action.

Lemma 7. *Every type τ has the form:*

$$\tau \equiv \tau_1 \to \cdots \to \tau_m \to e$$

where $m \geq 0$ and e is an atomic type.

Proof. (Induction in τ size)

- if $\tau \equiv a$ then $m = 0$;
- if $\tau \equiv \sigma \to \rho$, by IH $\sigma \equiv \sigma_1 \to \cdots \to \sigma_n \to e_\sigma$ and $\rho \equiv \rho_1 \to \cdots \to \rho_k \to e_\rho$, and then $m = n+k+1$

$$\tau \equiv (\sigma_1 \to \cdots \to \sigma_n \to e_\sigma) \to \rho_1 \to \cdots \to \rho_k \to e_\rho$$

□

Definition 6. *For $\tau \equiv \tau_1 \to \cdots \to \tau_m \to e$, the following definitions are convenient:*

*(i) The occurrences of $\tau_1, ..., \tau_n$ and e will be called **premises** and **conclusion** (or **tail**) of τ;*

*(ii) m will be called the **arity** of τ;*

*(iii) Two type-occurrences will be called **isomorphic** iff they are occurrences of the same type;*

(iv) If the tail-components of σ and τ are isomorphic we may say:

$$Tail(\sigma) \cong Tail(\tau)$$

If we have that a β-nf is long, some interesting things happens with indexes in the form given in the lemma 2.

Comment 1. *Let τ be any type; say τ has form*

$$\tau \equiv \tau_1 \to \cdots \to \tau_m \to e \quad (m \geq 0, e \text{ an atom})$$

and let M^τ be any β-nf with type τ. By the lemma 2, M^τ has form

$$\lambda x_1^{\tau_1} ... x_k^{\tau_k} . (v^{(\rho_1 \to \cdots \rho_n \to \tau^*)} M_1^{\rho_1} \cdots M_n^{\rho_n})^{\tau^*})^{(\tau_1 \to \cdots \tau_k \to \tau^*)}$$

where $0 \leq k \leq m$ and $\tau^* \equiv \tau_{k+1} \to \cdots \to \tau_m \to e$.

If $M^\tau \in Long(\tau)$, then

(i) $k = m$ and $\tau^* \equiv e$

(ii) the types of $x_1, .., x_m$ coincide with the premises of τ

(iii) the tail of the type of v is isomorphic to $Tail(\tau)$

(iv) if M^τ is closed then $m \geq 1$ and v is an x_i $(1 \leq i \leq m)$ and

$$\tau_i \equiv \rho_1 \to \cdots \to \rho_n \to e$$

Example 3 (A type τ with $\#(\tau) = 1$)**.** *$\tau \equiv (a \to b \to c) \to (a \to b) \to c$ has exactly one normal inhabitant:*

$$\boldsymbol{S}^\tau \equiv \lambda x^{a \to b \to c}\, y^{a \to b}\, z^a \cdot x\, z\, (y\, z) \text{ is its normal inhabitant and is} \in Long(\tau) \cap Princ(\tau)$$

Proof. Follow the steps of application of the algorithm:

Step 1. Start by proving that $Long(\tau) = \{\mathbf{S}^\tau\}$, and looking at the structure of τ; by following the previous comments; we must have:

$$\begin{cases} m &= 3 \\ e &\equiv c \\ \tau_1 &\equiv a \to b \to c \\ \tau_2 &\equiv a \to b \\ \tau_3 &\equiv a \end{cases}$$

So the searched term is

$$M^\tau \equiv (\lambda x_1^{\tau_1}\, x_2^{\tau_2}\, x_3^{\tau_3} \cdot (v^{(\rho_1 \to \cdots \rho_n \to c)}\, M_1^{\rho_1} \cdots M_n^{\rho_n})^c)^{(\tau_1 \to \tau_2 \to \tau_3 \to c)} \in Long(\tau)$$

By the item (iv) of the previous comments, v must be one of x_1, x_2, x_3 whose type's tail is isomorphic to $Tail(\tau) \equiv c$. Then the unique possibly choose is $v \equiv x_1$ followed by exactly two arguments. Hence M must have the form:

$$M \equiv \lambda x_1^{a \to b \to c}\, x_2^{a \to b}\, x_3^a \cdot x_1^{a \to b \to c}\, U^a\, V^b \tag{1}$$

Step 2. Searching for suitables U^a and V^b we have, by the comments(i), that $U^a \equiv (w\, P_1 \cdots P_r)^a$ $(r \geq 0)$. And w is an x_i whose type's tail is isomorphic to the tail of the type of U^a. This tail is an occurrence of a, so $w \equiv x_3$ since x_3 has no premises so $r = 0$ and

$$U^a \equiv x_3^a \tag{2}$$

Because b is an atom, V^b cannot be an abstracted. Moreover its head must be x_i whose type's tail is an occurrence of b and the only possibility is x_2, so

$$V^b \equiv x_2^{a \to b}\, W^a \tag{3}$$

Step 3. Just as for U^a, the only possibility for W^a is

$$W^a \equiv x_3^a \tag{4}$$

And now we have the conclusion (modulo $\equiv_\alpha$):

$$\mathbf{S}^\tau \equiv \lambda x_1^{a \to b \to c}\, x_2^{a \to b}\, x_3^a \cdot x_1\, x_3\, (x_2\, x_3) \in Long(\tau)$$

By the lemma 4 and the fact that $\mathbf{S}^\tau$ is a $\beta\eta$-nf: $Nhabs(\tau) = \{S^\tau\}$ and, by the **PT algorithm** τ is a principal type of $\mathbf{S}^\tau$: $Nprinc(\tau) = \{S^\tau\}$ □

Example 4 (A type τ with $\#(\tau) = 0$)**.** *No type that is a skeleton has inhabitants.*

Proof. Let $\tau \equiv \tau_1 \to ... \to \tau_m \to e$ $(m \geq 1)$ be a skeleton. If τ had inhabitants it would have at least one long normal one, by the lemma 3, and this inhabitant has the form

$$\lambda x_1 ... x_m \cdot x_i\, M_1 \cdots M_n$$

There is an x_i whose type is an occurrence of e, but τ is skeletal so e cannot occur in any τ_i. Hence τ has no inhabitants. □

Example 5 (Pierce's Law)**.** $\tau \equiv ((a \to b) \to a) \to a$ *has no inhabitant.*

Proof. If $M^\tau \in Long(\tau)$ then

$$M^\tau \equiv \lambda x^{(a \to b) \to a} \cdot v\, U_1 \cdots U_n \quad (n \geq 0)$$

with $v \equiv x$, since M^τ is closed. Hence $n = 1$

$$M^\tau \equiv \lambda x^{(a\to b)\to a} \cdot x^{(a\to b)\to a}\, U^{a\to b}$$

Since $a \to b$ has just one premise:

$$U^{a\to b} \equiv \lambda y^a \cdot (w\, V_1 \cdots V_r)^b \quad (r \geq 0)$$

And M^τ is closed so w must be x or y, at the same that it has a type whose tail is an occurrence of b, but neither x nor y has such type ($Long(\tau) = \emptyset$). Then we conclude, by the remarks and the lemma 3 that

$$Long(\tau) = \emptyset \Rightarrow Nhabs(\tau) = \emptyset \Rightarrow Habs(\tau) = \emptyset$$

□

Example 6 (A type τ with $\#(\tau) = m$). *For $m \in \mathbb{N}^*$, $\tau \equiv a \to \cdots \to a$ $(m+1$ a's$)$ has m inhabitants.*

Proof. Any $M^\tau \in Long(\tau)$ must have form

$$M^\tau \equiv \lambda x_1^a \ldots x_m^a \cdot v\, V1 \cdots V_n$$

$v \equiv x_i$ for some $i \leq m$, but the types of $x_1, \ldots, x_m$ have no premises, so $n = 0$. Hence

$$M^\tau \equiv \lambda x_1^a \ldots x_m^a \cdot x_i^a,$$

this kind of term is called **selector** or **projector** and

$$Long(\tau) \equiv Nprinc(\tau) \equiv \{\lambda x_1^a \ldots x_m^a \cdot x_i^a \mid 1 \leq i \leq m\}$$

□

Example 7 (Some other interesting examples). *Follow the example of the main combinators ($\boldsymbol{B}$, $\boldsymbol{C}$, $\boldsymbol{K}$, $\boldsymbol{I}$, $\boldsymbol{W}$ and the Church numerals $\bar{n}$):*

	Type τ	$Nhabs(\tau)$	$Long(\tau)$	$Nprinc(\tau)$
(i)	$a \to a$	$\boldsymbol{I}$	$\boldsymbol{I}$	$\boldsymbol{I}$
(ii)	$a \to b \to a$	$\boldsymbol{K}$	$\boldsymbol{K}$	$\boldsymbol{K}$
(iii)	$(a \to b) \to (c \to a) \to c \to b$	$\boldsymbol{B}$	$\boldsymbol{B}$	$\boldsymbol{B}$
(iv)	$(a \to b \to c) \to b \to a \to c$	$\boldsymbol{C}$	$\boldsymbol{C}$	$\boldsymbol{C}$
(v)	$(a \to a \to b) \to a \to b$	$\boldsymbol{W}$	$\boldsymbol{W}$	$\boldsymbol{W}$
(vi)	$(a \to a) \to a \to a$	$\boldsymbol{I}, \bar{0}, \bar{1}, \bar{2}, \ldots$	$\bar{0}, \bar{1}, \bar{2}, \ldots$	$\bar{2}, \ldots$

$$\begin{array}{rclrcl} \boldsymbol{B} & \equiv & \lambda xyz \cdot x(yz) & \boldsymbol{C} & \equiv & \lambda xyz \cdot xzy \\ \boldsymbol{K} & \equiv & \lambda xy \cdot x & \boldsymbol{I} & \equiv & \lambda x \cdot x \\ \boldsymbol{W} & \equiv & \lambda xy \cdot xyy & \bar{n} & \equiv & \lambda xy \cdot x^n y \end{array}$$

Proof. Follow the proofs

(i) $M^\tau \equiv \lambda\, x_1^a \cdot x_1^a\, V_1 \cdots V_n$, but x_1^a admits no argument so $\{M^\tau \equiv \lambda\, x_1 \cdot x_1\} \equiv Long(\tau)$ (a particular case of the example 6, when $m = 1$);

(ii) $M^\tau \equiv \lambda\, x_1^a\, x_2^b\, x_1^a\, V_1 \cdots V_n$, but x_1^a admits no argument so $\{M^\tau \equiv \lambda\, x_1^a\, x_2^b\, x_1^a\} \equiv Long(\tau)$;

(iii) $M^\tau \equiv \lambda\, x_1^{a\to b}\, x_2^{c\to a}\, x_3^c \cdot x_1^{a\to b}\, U^a$, but $U^a \equiv x_2^{c\to a}\, V^c$ and $V^c \equiv x_3^c$, though $\{M^\tau \equiv \lambda\, x_1^{a\to b}\, x_2^{c\to a}\, x_3^c \cdot x_1^{a\to b}\, (x_2^{c\to a}\, x_3^c)\} \equiv Long(\tau)$;

(iv) $M^\tau \equiv \lambda\, x_1^{a\to b\to c}\, x_2^b\, x_3^a \cdot x_1\, U^a\, V^b$, but $U^a \equiv x_3^a$ and $V^b \equiv x_2^b$, though $\{M^\tau \equiv \lambda\, x_1^{a\to b\to c}\, x_2^b\, x_3^a \cdot x_1^{a\to b\to c}\, x_3^a\, x_2^b\} \equiv Long(\tau)$;

(v) $M^\tau \equiv \lambda\, x_1^{a\to a\to b}\, x_2^a \cdot x_1^{a\to a\to b}\, U^a\, V^a$, but $U^a \equiv V^a \equiv x_2$, though $\{M^\tau \equiv \lambda\, x_1^{a\to a\to b}\, x_2^a \cdot x_1^{a\to a\to b}\, x_2^a\, x_2^a\} \equiv Long(\tau)$;

(vi) $M^\tau \equiv \lambda\, x_1^{a\to a}\, x_2^a \cdot x_1^{a\to a}\, V_1^a$ or $M^\tau \equiv \lambda\, x_1^{a\to a}\, x_2^a \cdot x_2^a$, but $V_1^a \equiv x_1^{a\to a}\, V_2^a$ or $V_1^a \equiv x_2^a$,
and this process can be done indefinitely to conclude that
$Long(\tau) \equiv \{\lambda\, x_1\, x_2 \cdot x_2 \equiv \bar{0},\, \lambda\, x_1\, x_2 \cdot x_1\, x_2 \equiv \bar{1},\, \lambda\, x_1\, x_2 \cdot x_1\,(x_1\, x_2) \equiv \bar{2},\, ...,\, \lambda\, x_1\, x_2 \cdot x_1^n\, x_2 \equiv \bar{n},\, ...\}$

An important observation is that in the items (i) to (v) $Nhabs(\tau) \equiv Long(\tau) \equiv Nprinc(\tau)$, but in (vi) we have a trivial inhabitant in $Nhabs(\tau) - Long(\tau)$ not reached by the algorithm (the identity **I**). Moreover one could check in the same item (by the **PT algorithm**) that the type τ is not a principal type for $\bar{0}$ and $\bar{1}$ but only for $\bar{2}$, $\bar{3}$, ... If $n \geq 2$ it forces the arguments of the principal type being an a occurrence of the type a. □

4 The search and the counting algorithms

Now, after a quick definitions of $\mathbb{D}(\tau)$ and $Depth(\tau)$, we are only enunciating these two lemmas (The stretching and the shrinking lemma) and their proofs will be given to the end of the section 5.

Definition 7. $\mathbb{D}(\tau) := |\tau| \times ||\tau||$. *Where $|\tau|$ is the number of atom occurrences and $||\tau||$ is the number of distinct atom in τ.*

Definition 8. *The depth of typed or untyped terms is given by:*

(i) $Depth(\lambda x_1 \cdots x_m\, .\, y) = 0$

(ii) $Depth(\lambda x_1 \cdots x_m\, .\, y\, M_1 \cdots M_n) = 1 + \underset{1\leq j\leq n}{Max}\, Depth(M_j)$ *if* $n > 0$

Notation 4. *The sets of all long normal inhabitants of τ (typed or untyped) with Depth $\leq d$ will both be called $Long(\tau, d)$.*

Lemma 8 (Stretching Lemma)**.** *If $Long(\tau)$ has a member M^τ with depth $d \geq \|\tau\|$ then it has members depths greater than any given integer, and hence is infinite.*

Lemma 9 (Shrinking Lemma)**.** *If $Long(\tau)$ has a member M^τ with depth $\geq \mathbb{D}(\tau)$ then it has a member N^τ with:*

$$\mathbb{D}(\tau) - \|\tau\| \leq Depth(N^\tau) < \mathbb{D}(\tau)$$

Colollary 1. *If $Long(\tau)$ has a member M^τ with depth $\geq (\tau)$ then it has a member N^τ with*

$$\|\tau\| \leq Depth(N^\tau) < \mathbb{D}(\tau)$$

Proof. If $Long(\tau)$ has a member then τ is composite by the lemma 4. Hence $|\tau| \geq 2$, so

$$\mathbb{D}(\tau) - \|\tau\| \geq 2\|\tau\| - \|\tau\| = \|\tau\|$$

and by the lemma 9

$$\|\tau\| \leq \mathbb{D}(\tau) - \|\tau\| \leq Depth(N^\tau) < \mathbb{D}(\tau)$$

□

Lemma 10. $Depth(M) < |M|$

Proof. Let $M \equiv \lambda x_1 \cdots x_n\, .\, y M_1 \cdots M_m$. The induction is in m and in $|M|$.

(IB) : $m = 0$, so $M \equiv \lambda x_1 \cdots x_n\, .\, y$. $Depth(\lambda x_1 \cdots x_n\, .\, y) = 0 < n + 1 = |\lambda x_1 \cdots x_n\, .\, y|$.

(IH) : By induction hypothesis:

- $Depth(\lambda x_1 \cdots x_n\, .\, y M_1 \cdots M_k) = 1 + Max\{Depth(M_j)\,|\, 1 \leq j \leq k\} < |\lambda x_1 \cdots x_n\, .\, y M_1 \cdots M_k|$;
- $Depth(M_{k+1}) < |M_{k+1}|$.

But,

- $Depth(\lambda x_1 \cdots x_n\, .\, y M_1 \cdots M_{k+1}) = 1 + Max\{Depth(M_j)\,|\, 1 \leq j \leq k+1\}$ and

- $1 + Max\{Depth(M_j) \,|\, 1 \leq j \leq k+1\} < |\lambda x_1 \cdots x_n \,.\, yM_1 \cdots M_k| + |M_{k+1}|$ and
- $|\lambda x_1 \cdots x_n \,.\, yM_1 \cdots M_k| + |M_{k+1}| = |\lambda x_1 \cdots x_n \,.\, yM_1 \cdots M_k M_{k+1}|$

Thus,

$Depth(\lambda x_1 \cdots x_n \,.\, yM_1 \cdots M_{k+1}) < |\lambda x_1 \cdots x_n \,.\, yM_1 \cdots M_k M_{k+1}|$.

□

Definition 9 (nf-schemes)**.** *A **nf-scheme** is a β-nf that may contain meta-variables under restrictions:*

(i) each nf-scheme is a β-nf without bound-variables clashes;

(ii) meta-variables do not bind (λV is forbidden);

(iii) in a composite nf-scheme meta-variables only occur in argument positions;

(iv) each meta-variable in a nf-scheme occurs only once.

Definition 10 (Proper nf-scheme)**.** *A **proper nf-scheme** is a nf-scheme that contain at least one meta-variable.*

Lemma 11 (Search Theorem for $Long(\tau)$)**.** *The search algorithm below accepts any composite type τ and outputs a finite or infinite sequence of sets $\mathcal{A}(\tau, d)(d = 0, 1, 2, 3, \cdots)$,*

(i) each member of $\mathcal{A}(\tau, d)$ is a closed long typed nf-scheme with type τ, and is either

- *a proper nf-scheme with depth d, or*
- *a term with depth $d - 1$*

(ii) $\mathcal{A}(\tau, d)$ is finite.

(iii) $Long(\tau, d) \subseteq \mathcal{A}(\tau, 0) \cup \cdots \cup \mathcal{A}(\tau, d)$

(iv) If the set of all terms in $\mathcal{A}(\tau, d)$ is called $\mathcal{A}_{terms}(\tau, d)$, then :

$$Long(\tau) = \bigcup_{d \geq 0} \mathcal{A}_{terms}(\tau, d)$$

Proof. The proof of *(i)* and *(ii)* is using induction in d.

(IB) The induction base : if $d = 0$, then $\mathcal{A}(\tau, d) = V^\tau$ or $\mathcal{A}(\tau, d) = \emptyset$. In any way, each element of $\mathcal{A}(\tau, d)$ has depth 0 and the set is finite.

(IH) Assume, by induction hypothesis, that for all k less or equal than d :

(i) $\mathcal{A}(\tau, k)$ is finite

(ii) each member of $\mathcal{A}(\tau, k)$ is a closed long

(iii) each member of $\mathcal{A}(\tau, k)$ is a proper nf-scheme with depth k or a term with depth $k - 1$

The goal is proof that the same properties hold for $\mathcal{A}(\tau, d+1)$. In other to do it, it will be analyzed each case of algorithm in the computation of $\mathcal{A}(\tau, d+1)$.

(Case 1) $\mathcal{A}(\tau, d) = \emptyset$ or no member of $\mathcal{A}(\tau, d)$ contains a meta-variable.

If $\mathcal{A}(\tau, d) = \emptyset$, then, by vacuity, the properties hold.

If no member of $\mathcal{A}(\tau, d)$ contains a meta-variable, the answer is the sequence $\mathcal{A}(\tau, 0), \cdots, \mathcal{A}(\tau, d)$, however, the answer is undefined.

(Case 2) Take one member of $\mathcal{A}(\tau, d)$. By IH, it is a proper nf-scheme with depth d or a term with depth $d-1$. Suppose that this member is a proper nf-scheme. Any suitable replacement of it has the form $\lambda x_1 \cdots x_n \cdot (hV_1 \cdots V_m)$. If $m = 0$ for all of that, they must have the form $\lambda x_1 \cdots x_n \cdot h$ and the result is a closed lambda term. The depth of all suitable replacement is 0 and the depth of the closed lambda, that belongs to $\mathcal{A}(\tau, d+1)$, term is d. In the other hand, if $m > 0$, the depth of $\lambda x_1 \cdots x_n \cdot (hV_1 \cdots V_m)$ is 1. Thus, after all the replacements of the meta-variables, the depth of the proper nf-scheme increases 1. Therefore, the result is a nf-scheme with depth $d+1$, that belongs to $\mathcal{A}(\tau, d+1)$. The number of replacements is finite, thus, the number of elements in $\mathcal{A}(\tau, d+1)$ is also finite.

The item *(iii)* states that the algorithm is complete and its proof is postponed to the lemma 16. The Part *(iv)* follows easily from *(i)-(iii)*. □

Definition 11. *For the sets $\mathcal{A}(\tau, d)$ and $\mathcal{A}_{terms}(\tau, d)$, define:*

$$\begin{array}{lcl} \mathcal{A}(\tau, \leq d) & = & \mathcal{A}(\tau, 0) \cup ... \cup \mathcal{A}(\tau, d) \\ \mathcal{A}_{terms}(\tau, \leq d) & = & \mathcal{A}_{terms}(\tau, 0) \cup ... \cup \mathcal{A}_{terms}(\tau, d) \end{array}$$

Definition 12 (The search algorithm $\mathcal{A}(\tau, n)$)**.** *input: any type τ*
If τ is an atom, it as no inhabitant.
Otherwise,

1. *Chose a meta-variable V :*

$$A(\tau, 0) = V^\tau$$

$d+1$ *Assume that $A(\tau, d)$ is defined.*

(a) *If $A(\tau, d) = \emptyset$ or no member of $A(\tau, d)$ contains meta-variable, then stop. $A(\tau, d+1)$ is undefined, and the algorithm outputs the sequence $A(\tau, 1), \cdots, A(\tau, d)$.*

(b) *Otherwise, apply the following steps:*

i. *Given any proper $X^\tau \in A(\tau, d)$, list the meta-variables of X^τ:*

$$V_1^{\rho_1}, \cdots, V_n^{\rho_n}$$

Apply the following steps to each meta-variable:

A. *Given any meta-variable V^ρ in $X^\tau \in A(\tau, d)$, say :*

$$\rho \equiv \sigma_1 \to \cdots \to \sigma_m \to a$$

B. *List all σ_j such that $tail(\sigma_j) \cong a \cong tail(\rho)$. Each σ_j has the form:*

$$\sigma_j \equiv \sigma_{j,1} \to \cdots \to \sigma_{j,n_j} \to a$$

Thus, define:

$$Y_j^\rho \equiv \lambda x_1^{\sigma_1} \cdots x_m^{\sigma_m} \cdot (x_j^{\sigma_j} V_{j,1}^{\sigma_{j,1}} \cdots V_{j,n_j}^{\sigma_{j,n_j}})^a$$

C. *List the abstractors that cover the occurrence of V^ρ in X^τ in order that they occur. They are:*

$$\lambda z_1^{\zeta_1} \cdots z_t^{\zeta_t}$$

List each ζ_i such that $tail(\zeta_i) = a$. It has the form:

$$\zeta_i \equiv \zeta_{i,1} \to \cdots \to \zeta_{i.h_i} \to a$$

Define

$$Z_i^\rho \equiv \lambda x_1^{\sigma_1} \cdots x_m^{\sigma_m} \cdot (z_i^{\zeta_i} V_{i,1}^{\zeta_{i,1}} \cdots V_{i,h_i}^{\zeta_{i,h_i}})^a$$

ii. *When the last step was applied in all meta-variables in X^τ, the result is a list of suitable replacements of V_i in X^τ. If at least one of $V_1 \cdots V_q$ has no suitable replacements, abandon X^τ, call it a reject, and start applying step b to the next member of $A(\tau, d)$.*
If all $V_1 \cdots V_q$ has suitable replacements, X^τ is called extendable. In this case, list all possible replacements:

$$\langle W_1^{\rho_1}, \cdots, W_q^{\rho_q} \rangle$$

Where each $W_j^{\rho_j}$ is a suitable replacement for V_j. For each element of sequence, construct a new nf-scheme $X^{*\rho}$ from X^{ρ}, replacing V_j for $W_j^{\rho_j}$.

(c) If $A(\tau, d)$ contains at least one nf-scheme, define $A(X^{\tau}, d+1)$ contain all the extensions of all extendable proper of nf-schemes in $A(\tau, d)$

Example 8. *Apply the search algorithm for the following type:*

$$\tau \equiv (a \to b \to c) \to (a \to b) \to a \to c$$

Consider $SR(V^{\sigma})$ as a list of suitable replacements of V^{σ}. The sets are:

$$\mathcal{A}(\tau, 0) = \{V_1^{\tau}\}$$

$$SR(V_1^{\tau}) = \langle \lambda x_1^{a\to b\to c} x_2^{a\to b} x_3^{a} \cdot x_1^{a\to b\to c} V_2^{a} V_3^{b}\rangle$$

$$\mathcal{A}(\tau, 1) = \{\lambda x_1^{a\to b\to c} x_2^{a\to b} x_3^{a} \cdot x_1^{a\to b\to c} V_2^{a} V_3^{b}\}$$

$$SR(V_2^{a}) = \langle x_3^{a}\rangle$$

$$SR(V_3^{b}) = \langle x_2^{a\to b} V_4^{a}\rangle$$

$$\mathcal{A}(\tau, 2) = \{\lambda x_1^{a\to b\to c} x_2^{a\to b} x_3^{a} \cdot x_1^{a\to b\to c} x_3^{a} x_2^{a\to b} V_4^{a}\}$$

$$SR(V_4^{a}) = \langle x_3^{a}\rangle$$

$$\mathcal{A}(\tau, 3) = \{\lambda x_1^{a\to b\to c} x_2^{a\to b} x_3^{a} \cdot x_1^{a\to b\to c} x_3^{a} x_2^{a\to b} x_3^{a}\}$$

Example 9. *Apply the search algorithm for the following type:*

$$\tau \equiv a \to \cdots \to a \to a$$

Where $|\tau| = m + 1$. The sets are:

$$\mathcal{A}(\tau, 0) = \{V_1^{\tau}\}$$

$$SR(V_1^{\tau}) = \langle (\lambda x_1^{a} \cdots x_{m+1}^{a} \cdot x_1^{a}), \cdots, (\lambda x_1^{a} \cdots x_{m+1}^{a} \cdot x_{m+1}^{a})\rangle$$

$$\mathcal{A}(\tau, 1) = \{(\lambda x_1^{a} \cdots x_{m+1}^{a} \cdot x_1^{a}), \cdots, (\lambda x_1^{a} \cdots x_{m+1}^{a} \cdot x_{m+1}^{a})\}$$

Example 10. *Apply the search algorithm for the following type:*

$$\tau \equiv ((a \to b) \to a) \to a$$

The sets are:

$$\mathcal{A}(\tau, 0) = \{V_1^{\tau}\}$$

$$SR(V_1^{\tau}) = \langle \lambda x_1^{(a\to b)\to a}.V_2^{a}\rangle$$

$$\mathcal{A}(\tau, 1) = \{\lambda x_1^{(a\to b)\to a}.V_2^{a}\}$$

$$SR(V_2^{a}) = \langle x_1^{(a\to b)\to a} V_3^{a\to b}\rangle$$

$$\mathcal{A}(\tau, 2) = \{\lambda x_1^{(a\to b)\to a}.x_1^{(a\to b)\to a} V_3^{a\to b}\}$$

$$SR(V_3^{a\to b}) = \langle\rangle$$

$$\mathcal{A}(\tau, 3) = \emptyset$$

Definition 13 (Counting Algorithm for *Long*(τ))**.** *In τ is an atom, Long(τ) is empty by the lemma 4. If τ is composite, apply the search algorithm (definition 12) to τ; this outputs a finite or infinite sequence of sets*

$$\mathcal{A}(\tau, d) \quad (d = 0, 1, 2, ...)$$

Stop the search algorithm at $d = \mathbb{D}(\tau)$ *and enumerate* $\mathcal{A}_{terms}(\tau, \leq \mathbb{D}(\tau))$.

[By the lemma 11, if $Long(\tau)$ had a member with depth $\geq \mathbb{D}(\tau)$
it would have one with depth $< \mathbb{D}(\tau)$.]

Case I: $\mathcal{A}_{terms}(\tau, \leq \mathbb{D}(\tau)) = \emptyset$.

[By the lemma 9, if $Long(\tau)$ had a member with depth $\geq \mathbb{D}(\tau)$
it would have one with depth $< \mathcal{D}(\tau)$.]

Case II: $\mathcal{A}_{terms}(\tau, \leq \mathbb{D}(\tau))$ *has a member with depth* $\geq \|\tau\|$. *Then by the lemma 8* $Long(\tau)$ *is infinite. To enumerate* $Long(\tau)$, *apply the search algorithm to enumerate* $\mathcal{A}_{terms}(\tau, d)$ *for* $d = 0, 1, 2, ...$

[By the lemma 11(iv) the union of these sets is $Long(\tau)$].

Case III: $\mathcal{A}_{terms}(\tau, \leq \mathbb{D}(\tau))$ *has members but they all have depth* $< \|\tau\|$. *Then* $Long(\tau) = \mathcal{A}_{terms}(\tau, \leq \mathbb{D}(\tau))$, *which is finite.*

[The only way for $Long(\tau)$ to differ from this set would be for $Long(\tau)$ to
have members with depth $d \geq \mathbb{D}(\tau)$), but by the lemma 1 it would then have a member
with $\|\tau\| \leq d < \mathbb{D}(\tau)$) contrary to the assumption of the present case.]

Theorem 1 (Counting long normal inhabitants)**.** *When given a type* τ *the Counting Algorithm outputs* $\#Long(\tau)$ *and an enumeration of* $Long(\tau)$.

Proof. The proof is in the above brackets with the algorithm's description. □

Colollary 2 (Counting $\beta\eta$-normal inhabitants)**.** *The algorithm of the definition 13 can be used to count and enumerate* $Nhabs_\eta(\tau)$ *for every* τ.

Proof. By 4 the members of $Nhabs_\eta(\tau)$ are the η-nf's of those $Long(\tau)$. And by the lemma 4(iii), $\#_\eta(\tau) = \#(Nhabs_\eta(\tau)) = \#(Long(\tau))$. □

Colollary 3 (Emptiness test)**.** *The algorithm of the definition 13 can be used to decide whether a type* τ *has no inhabitants.*

Proof. By the remark 1(i), $Habs(\tau) = \emptyset \Leftrightarrow \#(Nhabs(\tau)) = \#(\tau) = 0$. And by the lemma 4(ii) and (iii), $\#(\tau) = 0 \Leftrightarrow \#(Long(\tau)) = 0$. □

Definition 14 (Counting Algorithm for $Nhabs(\tau)$)**.** *If* τ *is an atom then* $Nhabs(\tau)$ *is empty by the lemma 4. If* τ *is composite, apply the algorithm of the definition 13 to count* $Long(\tau)$.

Case I*: If* $Long(\tau) = \emptyset$ *then* $Nhabs(\tau) = \emptyset$.

[By the lemma 3.]

Case II*: If* $Long(\tau) \neq \emptyset$ *then* $Nhabs(\tau)$ *is counted and enumerated by counting and enumerating* $Long(\tau)$ *and enumerating the* η*-family of each member of* $Long(\tau)$.

[By the lemma 4 these η-families are finite and their union is in $Nhabs(\tau)$.]

Theorem 2 (Counting all normal inhabitants)**.** *When given a type* τ *the algorithm of the definition 14 outputs* $\#(\tau)$ *and an enumeration of the set* $Nhabs(\tau)$ *of all* β*-normal inhabitants of* τ.

Proof. The proof is in the above brackets with the algorithm's description. □

Definition 15. *A type τ will be called **monatomic** iff $\|\tau\| = 1$, i.e. iff only one atom occurs in τ.*

Theorem 3. *Let τ be a monatomic type with the form $\tau \equiv \tau_1 \to ...\tau_m \to a$ ($m \geq 0$). Then*

(i) if at least one τ_i is composite, $\#(\tau)$ is either ∞ or 0;

(ii) if $\tau_1 \equiv ... \equiv \tau_m \equiv a$ then $\#(\tau) = m$.

Proof. The part (ii) is simple application of the lemma 6. To prove (i), assuming that $Nhabs(\tau)$ is finite and non-empty and that τ has form

$$\tau \equiv \tau_1 \to ... \to \tau_m \to a$$

with $m \geq 1$ and at least one premise composite. By the lemma 3 $Long(\tau)$ is also finite and has a member M^τ. By the lemma 8, $Depth(M^\tau) <\|\tau\| = 1$, so M^τ must have form

$$M^\tau \equiv \lambda x_1^{\tau_1} ... x_m^{\tau_m} \cdot x_p^{\tau_p} \quad (1 \leq p \leq m).$$

And $\tau_p \equiv a$ since M^τ is long. Choose a composite premise of τ; say it is τ_i and has form

$$\tau_i \equiv \tau_{i,1} \to ... \to \tau_{i,m_i} \to a \quad (m_i \geq 1).$$

For $j = 1, ..., m_i$ each τ_i, j has form

$$\tau_{i,j} \equiv \tau_{i,j,1} \to ... \to \tau_{i,j,m_{i,j}} \to a \quad (m_{i,j} \geq 0).$$

For each $j \leq m_i$ choose distinct new variables $y_1, ..., y_{m_{i,j}}$ and define

$$P_j^{\tau_{i,j}} \equiv \lambda y_1^{\tau_{i,j,1}} ... y_{m_{i,j}}^{\tau_{i,j,m_{i,j}}} \cdot x_p^a$$

where x_p^a is the rightmost variable in M^τ (and is therefore distinct from the y's). Then define

$$N^\tau \equiv \lambda x_1^{\tau_1} ... x_m^{\tau_m} \cdot (x_i^{\tau_i} P_1^{\tau_{i,1}} \cdots P_{m_i}^{\tau_{i,m_i}})^a$$

Clearly $N^\tau \in Long(\tau)$. But $Depth(N^\tau) \geq 1$ since N^τ contains $m_i \geq 1$ arguments. Hence $Long(\tau)$ is infinite by the lemma 8. □

5 Search-Completeness, Stretching and Shrinking lemmas

The aim of the this section is to prove two important lemmas that will serve as a basis for the proof of completeness of search and counting algorithms. The key role will lay the foundation by analyzing the structure of an arbitrary long typed nfscheme, and will be played by a slightly strengthened form of the subformula property. That property says in effect that the types of all the components of a closed β-nf M^τ are subtypes of τ. This implies that all the successes produced by the search algorithm, growing deeper and deeper, have the types of their components drawn from the same fixed finite set. This limitation is the source of the bounds in the stretching and shrinking lemmas.

We shall need the notation for positions, components and construction-trees. In writing positions a sequence of n 0's may be written as 0^n (with $0^0 = \emptyset$), and similarly for 1's and 2's. Remembering that every non-atomic nf-scheme X can be expressed uniquely in the form:

$$X \equiv \lambda x_1 ... x_m \cdots v\, Y_1 \cdots Y_n \quad (m + n \geq 1)$$

where v is one of $x_1, ..., x_n$ if X is closed. The construction-tree of such X is shown in Fig. 2. The **head** and **arguments** of X are respectively v and Y_1, ..., Y_n. We can note that the position of Y_i is

$$0^m\, 1^{n-i}\, 2 \quad (1 \leq i \leq n)$$

The Fig. 3 is presenting an example of construction tree of $\lambda x\, y \cdot x\, (\lambda u \cdot u\, V_1)\, V_2$.

Definition 16 (Typed nf-schemes, $\mathbb{TNS}(\Gamma)$)**.** ***Typed nf-schemes*** *are defined as $\mathbb{TT}(\Gamma)$ with meta-variables under the same restrictions pointed in the definition 9.*

Definition 17 (Long typed nf-schemes)**.** *A nf-scheme X^τ is long iff each component of X^τ with the form:*

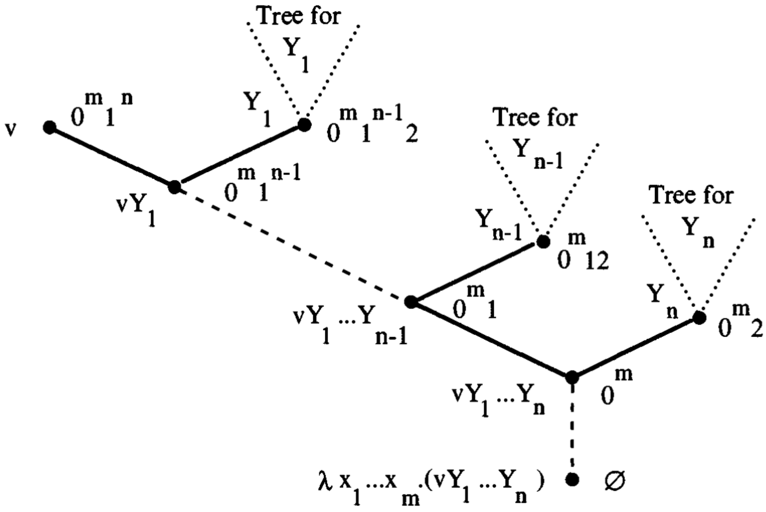


Figure 2: The construction-tree of a nf-scheme X.

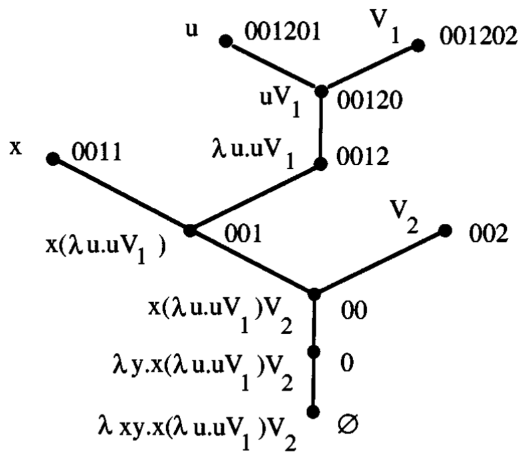


Figure 3: The construction-tree of $\lambda x\, y \cdot x\, (\lambda u \cdot u\, V_1)\, V_2$.

$$(z\, Y_1 \cdots Y_n)^\tau \quad (n \geq 0)$$

that is not in a function position has atomic type.

Example 11. *The nf-scheme (i) below is long, though in contrast the term (ii) is not:*

(i) $x^{(a\to b)\to c)}\, V^{a\to b}$,

(ii) $x^{(a\to b)\to c)}\, z^{a\to b}$.

Remembering that in (i) the meta-variable V *could be filled with* $(\lambda y^a \cdot z^b)^{a\to b}$.

Definition 18 (Subarguments)**.** *A **subargument** of a typed or untyped nf-scheme* X *is a component that is an argument of* X *or an argument of a proper component of* X.

Note 2. *The following notes will be useful:*

(i) All occurrences of meta-variables in a composite nf-scheme are subarguments.

(ii) A subargument of a subargument of X *is a subargument of* X.

Lemma 12. *A component* Y *of a typed or untyped nf-scheme* X *is a subargument iff its position is not* $\emptyset$ *and the last symbol in its position is* 2.

Proof. Induction in $|X|$:

Every non-atomic nf-scheme has the form

$$X \equiv \lambda x_1 \dots x_m \cdots v\, Y_1 \cdots Y_n \quad (n+m \geq 1);$$

where each

$$Y_i \equiv z_i\, U_1 \cdots U_k \quad (k \geq 0).$$

So following the Fig 2:

(IB) If a given Y_i is a meta-variable his position is $\neq \emptyset$ and it will be $0^m 1^{(n-i)} 2$;

(IS) And if Y_i is not a meta-variable, by IH, his subarguments have positions $\neq \emptyset$ and ended in 2. Then because the subarguments of Y_i are subarguments of X we must have that all subarguments of X are $\neq \emptyset$ and ended in 2.

□

Definition 19 (Relative depth)**.** *The* 2-***length*** *of a position-string* p *is the number of* 2*'s in* p*. The* ***depth in*** X *of a subargument* Z *of* X *is the* 2*-length of its position (i.e. the number of right-hand choices made when traveling up the tree of* X *from the bottom node to* Z *(cf. Fig. 2)).*

Lemma 13. *Let* X *be a typed or untyped nf-scheme with* $Depth(X) \geq 1$*. Then*

(i) $Depth(X)$ *is the maximum of the depths in* X *of all subarguments of* X*,*

(ii) X *has at least one subargument whose depth in* X *is the same as* $Depth(X)$ *and each such subargument is an atom or abstract atom.*

Proof. (Induction in $|X|$)

Given $X \equiv \lambda x_1 \dots x_m \cdots v\, Y_1 \cdots Y_n$ $(n+m \geq 1)$, by the definition 8 we have that:

(IB) if $n = 0$ then $Depth(X) = Depth(\lambda x_1 \dots x_m \cdots v) = 0$

(IS) if $n \neq 0$ $Depth(\lambda x_1 \dots x_m \cdots v\, Y_1 \cdots Y_n) = 1 + \underset{1 \leq i \leq n}{Max}\, Depth(Y_i)$ But, by IH, for each Y_i we have:

- (i) $Depth(Y_i)$ is the maximum of the depths in Y_i of all subarguments of Y_i, and because of the fact that each subargument of Y_i is a subargument of X, we have that *(i)* holds;
- (ii) each Y_i has at least one subargument whose depth in Y_i is the same as $Depth(Y_i)$, so by the definition of *Depth*, X will have at least one subargument with depth equal to $Depth(X)$.

□

Definition 20 (Argument-branch)**.** *If* Z *is a subargument of a typed or untyped nf-scheme* X*, the* ***argument-branch from*** X ***to*** Z *is the sequence*

$$\langle Z_0, Z_1, \dots, Z_k \rangle \quad (k \geq 1)$$

such that $Z_0 \equiv X$ *and* Z_i *is an argument of* Z_{i-1} *for* $i = 1, \dots, k$*, and* $Z_k \equiv Z$*. It is called* ***unextendable*** *iff* Z *is an atom or abstract atom. Its* ***length*** *is* k *(not* $k+1$*).*

Lemma 14. *For any typed or untyped nf-scheme* X*:*

(i) the depth in X *of a subargument* Z *is the same as the length of the argument-branch from* X *to* Z*;*

(ii) $Depth(X)$ *is the maximum of the lengths of all argument-branches in* X*.*

Proof. For *(i)* induction in $|X|$

(IB) if $n = 0$ then $Depth(X) = Depth(\lambda x_1 \dots x_m \cdots v) = 0$ and the length to the argument-branch from X to X itself is 0.

(IS) if $n \neq 0$ $Depth(\lambda x_1 \ldots x_m \cdots v\, Y_1 \cdots Y_n) = 1 + \underset{1\leq i\leq n}{Max}\, Depth(Y_i)$, so by IH, the length argument-branch of a subargument Z_i of Y_i to Y_i will be the depth of Z_i in Y_i, and, by the definition of *Depth*, we conclude that length of the argument-branch to Z_i to X is equal to the depth of Z_i in X.

The item *(ii)* follows direct of the application of the lemma 13 in the item *(i)*. $\square$

Definition 21 (IA, CA). *Let Z be a subargument of typed or untyped nf-scheme X; say*

$$Z \equiv \lambda x_1 \ldots x_m \cdot y\, Z_1 \cdots Z_n \quad (m \geq 0, n \geq 0)$$

The ***Initial Abstractors sequence*** *$IA(Z)$ is the (possible empty) sequence*

$$IA(Z) = \langle x_1, \ldots, x_m \rangle$$

The ***Covering Abstractors sequence*** *$CA(Z, X)$ is defined to be $CA(Z, X) = \langle z_1, \ldots, z_Q \rangle$ where $\lambda z_1, \ldots, \lambda z_q$ are the abstractors in X whose scopes contain Z, written in the order they occur in X from left to right. Also define:*

$$Length(IA(Z)) = m, \qquad Length(CA(Z, X)) = q.$$

Note 3. *(i) If X has no bound-variable clashes the member of $IA(Z)$ are distinct and so are those of $CA(Z, X)$;*

(ii) $IA(Z)$ and $CA(Z, X)$ are sequences of variables not components

(iii) For typed nf-schemes each variable in $IA(Z)$ or $CA(Z, X)$ is typed;

(iv) If the argument-branch from X to Z is $\langle Z_0, \ldots, Z_k \rangle$ $(k \geq 1)$, then

$$CA(Z, X) = IA(Z_0) * \ldots * IA(Z_{k-1})$$

where "$*$" denotes concatenation of sequences. (Because the abstractors whose scopes contain Z are exactly the initial abstractors of $Z_0, \ldots, Z_{k-1}$.

Definition 22 (IAT). *Let Z^σ be a subargument of a typed nf-scheme X^τ; say*

$$Z^\sigma \equiv \lambda x_1^{\sigma_1} \ldots x_m^{\sigma_m} \cdot y\, Z_1 \cdots Z_n \quad (m \geq 0, n \geq 0)$$

The ***Initial Abstractors' Types sequence*** *$IAT(Z^\sigma)$ is defined to be*

$$IAT(Z^\sigma) = \langle \sigma_1, \ldots, \sigma_m \rangle$$

also define

$$Length(IAT(Z^\sigma) = m.$$

Definition 23 (Condensed tree of a type). *The* ***condensed construction-tree*** *of a type τ is defined by induction in $|\tau|$, thus. (Each of its nodes is labeled with a type ad a position.)*

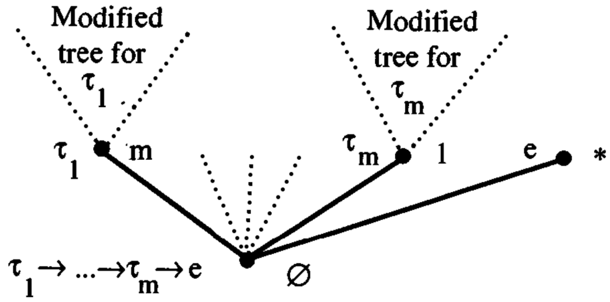
(i) If τ is an atom e, its condensed tree is a single node:

$$e \bullet \emptyset$$

(ii) If $\tau \equiv \tau_1 \to \ldots \tau_m \to e$ $(n \geq 1)$, construct its condensed tree from the condensed trees of $\tau_1, \ldots m, \tau_m$ by first replacing each position-label p in the tree of τ_i by $(m + 1 - i)p$ (for $i = 1, \ldots, m$), and combining the modified trees as follows:

Definition 24 (S-components). *Iff a node on the condensed tree of τ is labeled with a type σ and a position p we call the triple $\langle \sigma, p, \tau \rangle$ an* ***s-component*** *of τ.*

Definition 25 (Premises, tail). *If ρ is a a composite s-component of a type τ and $\rho \equiv \rho_1 \to \ldots \to \rho_m \to a$ $(m \geq 1)$, the s-components ρ_1, ..., ρ_m are called the* ***premises*** *of ρ and a is called the* ***conclusion*** *or* ***tail-component*** *of ρ.*



Definition 26 (Subpremises, subtails)**.** *An s-component of τ is called a* ***subpremise*** *of* ***subtail*** *of τ according as it is a premise or tail of another s-component of τ. The sets of all subpremises and all subtails of τ will be called, respectively,*

$$Subpremises(\tau), \quad Subtail(\tau)$$

Definition 27 (Positive and negative s-components)**.** *An s-component σ of τ is called* ***positive*** *or* ***negative*** *according as the number of non-asterisk symbols in tis position is even or odd. If σ is positive we say σ* ***occurs positively*** *in τ, otherwise σ* ***occurs negatively*** *in τ.*

Definition 28 ($NSS(\tau)$)**.** *If τ is composite, $NSS(\tau)$ is the set of all finite sequences $\langle \sigma_1, ..., \sigma_n \rangle$ ($n \geq 1$) such that τ contains a positive s-component with form*

$$\sigma_1 \to \ldots \to \sigma_n \to a$$

from some atom a. Each member of $NSS(\tau)$ is called a ***negative subpremise-sequence*** *(because it is a sequence of terms that have occurrences as negative subpremises in τ). The set of all members of the sequences in $NSS(\tau)$ will be called*

$$\bigcup NSS(\tau).$$

Lemma 15 (Enhanced Subformula Lemma)**.** *If Z^σ is a subargument of a closed long typed nf-scheme X^τ, then*

(i) σ occurs as a positive subpremise in τ,

(ii) if τ is an atom, $IAT(Z^\sigma) = \emptyset$,

(iii) if σ is composite, $IAT(Z^\sigma) \in NSS(\tau)$,

(iv) $NSS(\sigma) \subseteq NSS(\tau)$.

Proof. Since Z^τ is long, $IAT(Z^\tau)$ coincides with the sequence of all premises of σ, so (ii) holds. And also if τ is composite we have

$$IAT(Z^\tau) \in NSS(\sigma) \tag{5}$$

by the definition of $NSS(\sigma)$. Now (i) implies (iv), by the lemma **9E9.2**(iii), and (iv) and (5) imply (iii). Hence only (i) remains to be proved.

The proof of (i) is, again, an induction in $|X^\tau|$. So we shall prove

> If X^τ is a long member of $\mathbb{TNS}(\Gamma)$ and $\Gamma \equiv \{u_1 : \theta_1, \ldots u_p : \theta_p, V_1 : \phi_1, V_q : \phi_q\}$ and Z^σ is a subargument of X^τ, then σ occurs in a positive subpremise of
>
> $$\theta_1 \to \cdots \to \theta_p \to \tau \tag{6}$$

(IB) If X^τ is an atom the conclusion of (6) holds vacuously.

(IS) Let X^τ have form

$$(\lambda\, x_1^{\tau_1} \ldots x_m^{\tau^m} \cdot (y^{(\rho_1 \to \cdots \to \rho_n \to e)}\, X_1^{\rho_1} \cdots X_n^{\rho_n})^e)^{(\tau_1 \to \cdots \to \tau_m \to e)} \tag{7}$$

where $m, n \geq 0$ and $\tau \equiv \tau_1 \to \cdots \tau_m \to e$. Then either $y \equiv x_i$ for some $i \leq m$ or $y \equiv u_i$ for some $i \leq p$. If $y \equiv x_i$ then

$$\tau_i \equiv \rho_1 \to \cdots \to \rho_n \to e \tag{8}$$

and if $y \equiv u_i$ then

$$\theta_i \equiv \rho_1 \to \cdots \to \rho_n \to e \tag{9}$$

In both cases ρ_1, ..., ρ_n occurs as a positive subpremise of

$$\theta_1 \to \cdots \to \theta_p \to \tau \tag{10}$$

Now Z^σ must be in an $X_j^{\rho_j}$ for some $j \leq n$. If $Z^\sigma \equiv X_j^{\rho_j}$ then $\sigma \equiv \rho_j$ and the conclusion of (6) follows by the above considerations. Next, suppose Z^σ is a subargument of $X_j^{\rho_j}$. Note that

$$X_j^{\rho_j} \in \mathbb{TNS}(\{x_1 : \tau_1, ..., x_m : \tau_m\} \cup \Gamma) \tag{11}$$

Hence, by IH, σ occurs as a positive subpremise of

$$\tau_1 \to \cdots \to \tau_m \to \theta_1 \to \cdots \to \theta_p \to p_j$$

Thus σ occurs as a positive subpremise of (10), given (6). □

Colollary 4. *If X^τ is a closed long typed nf-scheme, the type of each meta-variable in X^τ either occurs as a positive subpremise of τ or is τ itself.*

Proof. Consequence of the lemmas 2(i) and 15(i). □

Colollary 5. *If X^τ is a closed long typed nf-scheme and Z^σ is a subargument of X^τ or $Z^\sigma \equiv X^\tau$, then*

(i) $Length(IA(Z^\sigma)) = Length(IAT(Z^\sigma)) \leq |\tau| - 1$,

(ii) $Length(CA(Z^\sigma, X^\tau)) \leq (|\tau| - 1) \times Depth(X^\tau)$
Further, if $\lambda v_1^{\rho_1}, ..., \lambda v_r^{\rho_r}$ are all the abstractors in X^τ (not just its initial ones), then

(iii) $\{\rho_1, ..., \rho_r\}$ *has at most* $|\tau| - 1$ *distinct members.*

Proof. Follow the proof

For *(i)*: $Length(IAT(Z^\sigma) \leq |\tau| - 1$ by the lemmas 15(iii) and **9E9.3(iv)**.

For *(ii)*: If $Z \equiv X$ the left side of (ii) is 0. If $Z \not\equiv X$ let $\langle Z_0, ..., Z_k\rangle$ $(k \geq 1)$ be the argument branch form X to Z; the, by the note 3(iv)

$$Length(CA(Z, X)) = Length(IA(Z_0)) + \cdots + Length(IA(Z_{k-1})) \leq k(|\tau| - 1)$$

by (i). But $Depth(X) \geq k$ by the lemma 14(ii), so (ii) holds.

For *(iii)*: Each ρ_i is in $IAT(X^\tau)$ or in $IAT(Y^\theta)$ for subargument Y^θ of X^τ; and in both cases $\rho_i \in \cup NSS(\tau)$ (in the former case trivially, and the later case follows by the lemmas 15(iii) and **9E9.3(iii)**.

□

Lemma 16 (Search-Completeness Lemma)**.** *Part (iii) of the theorem 11 holds; i.e. if τ is composite and $d \geq 0$, then*

$$Long(\tau, d) \subseteq \mathcal{A}(\tau, \leq d + 1).$$

Proof. The goal is to that $Long(\tau, d) \subseteq \mathcal{A}(\tau, \leq d + 1)$ holds. In order to do it, a property stronger will be proved. Define $\mathcal{L}^*(\tau, d)$ the set of long closed nf-schemes with depth d such that:

(a) All X^τ in $\mathcal{L}^*(\tau, d)$ is proper and its meta-variables have depth d.

(b) All subarguments with depth d in X^τ are meta-variables.

The news properties to prove are:

1. $\mathcal{L}^*(\tau, d) \subseteq \mathcal{A}(\tau, \leq d)$
2. $Long(\tau, d) \subseteq \mathcal{A}(\tau, \leq d+1)$

The properties are proved by induction in d:

(IB) Induction basis is $d = 0$. For property *(a)*, there is only a nf-scheme with a depth 0. This is a meta-variable. Thus:

$\{V^\tau\} = \mathcal{L}^*(\tau, d) \subseteq \mathcal{A}(\tau, \leq d) = \{V^\tau\}$.

For the property b, let $\tau \equiv \tau_1 \to \cdots \to \tau_m \to e$. Let M^τ a element of $Long(\tau, 0)$. It has the form: $\lambda y_1^{\tau_1} \cdots y_m^{\tau_m} \cdot y_i^{\tau_i}$. To construct $\mathcal{A}(\tau, 1)$, it outputs $\lambda y_1^{\tau_1} \cdots y_m^{\tau_m} \cdot y_i^{\tau_i}$ as suitable replacement of V^τ in $\mathcal{A}(\tau, 0)$, because τ is isomorphic to τ. Hence $\mathcal{A}(\tau, 1)$ must contains $\lambda y_1^{\tau_1} \cdots y_m^{\tau_m} \cdot y_i^{\tau_i}$.

(IS) Assume the properties *(a)* and *(b)* holds for d and prove that they will hold for $d+1$. Let $X \in \mathcal{L}^*(\tau, d+1)$. Then, $Depth(X) = d+1$ by lemma 13 X has a subargument whose depth in X is $d+1$, and by lemma 14 X has one whose depth is d. List all subargments: $W_1, \cdots, W_r$. Obviously, $W_1, \cdots, W_r$ are disjoint components since they have the same depth d in X. Since $Depth(X) = d+1$, $Depth(W_i) \leq 1$ for each i. X satisfies *(a)* and *(b)*, so no W_i is a meta-variable and it must has the form:

$W_i \equiv \lambda x_{i,1} \cdots x_{i,m_i} \cdot (y_i V_{i,1} \cdots V_{i,n_i})$.

Let be X‘ as a result of replace each meta-variable V_i in X by W_i. X‘ is a long and closed nf-scheme with depth d and it satisfies the condition *(b)* as a member of $\mathcal{L}^*(\tau, d)$. The condition *(b)* holds too, because if X‘ contained a meta-variable occurrence V at a depth $< d$ such a V could not be a V_i and hence would occur also in X at a depth $< d$ contrary to the assumption that X satisfies *(a)* relative to $d+1$.

Hence X‘ $\in \mathcal{L}^*(\tau, d)$. By induction hypothesis there is an X“ $\in \mathcal{A}(\tau, \leq d)$ that is alpha equivalent to X‘. Apply the step $d+1$ of search algorithm. It is easy to see that W_i is a suitable replacement for V_i. Hence the algorithm gives X as an extension of X“. So, $X \in \mathcal{A}(\tau, \leq d+1)$ giving the property 1 for $d+1$.

In other to prove property 2, let $M \in Long(\tau, d+1)$. The by 13. M has a subargument whose depth in M is $d+1$. List all subarguments without repetitions : $U_1, \cdots, U_r$. Clearly, $U_1, cdots, U_r$ are disjoints. Since $Depth(M) = d+1$, each U_i has depth 0. Therefore, it must have the form: $U_i \equiv \lambda x_{i,1} \cdots x_{i,m_i} \cdot y_i$. Let M‘ as a replacement of each meta-variable V_i by U_i in M. M‘ is a closed and long nf-scheme with depth $d+1$. Also, M‘ satisfies *(a) (b)* as a member of $\mathcal{L}^*(\tau, d+1)$ because all its meta-variables was introduced by the above replacements. Hence M‘ $\in \mathcal{L}(\tau, d+1)$

Therefore by property 1, there is an M“, differing from M‘ only for the renaming of meta-variables such that M“ $\in \leq d+1$. Apply the step $d+1$ of search algorithm in M“. Trivially, U_i is a suitable replacement for V_i in M“. Thus, the algorithm outputs M as a extension of M“. Therefore, $M \in \mathcal{A}(\tau, d+2)$.

□

Lemma 17 (Detailed Stretching Lemma). *If $Long(\tau)$ has a member M^τ with depth $\geq \|\tau\|$ then*

(i) there exists $M^{*\tau} \in Lon(\tau)$ with $Depth(M^{*\tau}) \geq Depth(M^\tau) + 1$,

(ii) $Long(\tau)$ is infinite.

Proof. *(ii)* follows from *(i)*. And to prove *(i)*, let M a typed closed long β-nf with type τ and without variable clashes. Let $d = Depth(M) \geq ||\tau|| > 1$. By the lemma 14, M has at least one argument-branch with length d. Choose any such branch, call it

$$\langle N_0, ..., N_d \rangle \tag{12}$$

where $N_0 \equiv M$ and N_{i+1} is an argument of N_i, for $i = 0, ..., d-1$. Each of $N_0, ..., N_d$ must have form

$$N_i \equiv \lambda\, x_{i,1} \ldots x_{i,m_i} \cdot y_i\, P_{i,1} \cdots P_{i,n_i} \quad (m_i, n_i \geq 0), \tag{13}$$

and for $i \leq d-1$ we have $N_{i+1} \equiv P_{i,k_i}$ for some $k_i < n_i$ (and hence $n_i \geq 1$). Since $d = Depth(M)$ the last member (12) has no arguments, so $n_d = 0$. For $i = 0, ..., d$ let B_i be the body of N_i; that is

$$B_i \equiv y_i\, P_{i,1} \cdots P_{i,n_i} \tag{14}$$

Since N_i is long, the type of B_i is an atom. And this atom occurs in τ by the lemma **2B3(ii)**. But $||\tau|| \leq d$ and there are $d+1$ components $B_0, ..., B_d$, at least two of these must have the same type. Choose any pair $\langle p, p+r \rangle$ such that B_p and B_{p+r} and

$$Depth(B_p) \geq r + Depth(B_{p+r}).$$

Define M^* to be the result replacing B_{p+r} in M by a copy of B_p (after changing bound variables in this copy to avoid clashes). Then M^* has an argument-branch with length $d+r$. (Its members are

$$N_0^*, ..., N_{p+r}^*, N_{p+1}^o, ..., N_d^o,$$

where for $i = 0, ..., p+r$ each N_i^* has the same position in M^* as N_i in M, and for $j = p+1, ..., d$ we have $N_j^o \equiv N_j$) hence by the lemma 14,

$$Depth(M^*) \geq d + r \geq d + 1$$

To complete the proof of *(i)* it only remains to show that M^* is a genuine typed term. This will be done by applying **lemma 5B2.1(ii)** on replacement in typed terms.

First, for $i = 0, ..., d$ let Γ_i be the context that assigns to the initial abstractors of N_i i the types they have in M. Since M has no bound-variable clashes the variables in $\Gamma_0, ..., \Gamma_d$ are all distinct, so

$$\Gamma_0 \cup \cdots \cup \Gamma_d \text{ is consistent.} \tag{15}$$

Also every variable free in B_i B; is bound in one of $N_0, ..., N_i$ because M is closed and B_i is in N_i. Hence, by the definitions of typed term **(5A1)** and *Con* **(5A4)**,

$$B_i \in \mathbb{TT}(\Gamma_0 \cup \cdots \cup \Gamma_i), \quad Con(B_i) \subseteq \Gamma_0 \cup \cdots \cup \Gamma_i \tag{16}$$

To apply **5B2.1(ii)** it is enough to show that the set

$$Con(B_p) \cup Con(M) \cup \Gamma_0 \cup \cdots \cup \Gamma_{p+r} \tag{17}$$

is consistent. (The abstractors in M whose scopes contain B_{p+r} are exactly the initial abstractors of $N_0, ..., N_{p+r}$). But M is closed, so $Con(M) \equiv \emptyset$. And by (16),

$$Con(B_p) \subseteq \Gamma_0 \cup \cdots \cup \Gamma_p \subseteq \Gamma_0 \cup \cdots \cup \Gamma_{p+r}.$$

Then (17) is consistent by (15). □

Lemma 18 (Detailed Shrinking Lemma)**.** *If $Long(\tau)$ has a member M^τ with depth $\geq \mathbb{D}(\tau)$ then*

(i) it has a member $M^{*\tau}$ with

$$Depth(M^\tau) - \|\tau\| \leq Depth(M^{*\tau}) < Depth(M^\tau),$$

(ii) it has a member N^τ with

$$\mathbb{D}(\tau) - \|\tau\| \leq Depth(N^\tau) < \mathbb{D}(\tau)$$

Proof. Part *(ii)* is proved by repeating *(i)* and taking the first output with depth $< \mathbb{D}(\tau)$.

Part *(i)* is proved as follows. Let M be member of $Long(\tau)$ without bound-variable clashes. Let $d = Depth(M) \geq \mathbb{D}(\tau)$. In fact $\mathbb{D}(\tau) \geq 2$ because τ is composite e since atomic types have no inhabitants.

By the lemma 14, M has at least one argument-branch with length d; to reduce the depth of M we must shrink all these branches. Consider any such branch; just as in the proof of the stretching lemma 17 it has form

$$\langle N_0, ..., N_d \rangle \tag{18}$$

where $N_0 \equiv M$ and N_{i+1} is an argument of N_i, for $i = 0, ..., d-1$. And

$$N_i \equiv \lambda\, x_{i,1} \ldots x_{i,m_i} \cdot y_i\, P_{i,1} \cdots P_{i,n_i} \quad (m_i, n_i \geq 0), \tag{19}$$

and for $0 \leq i \leq d-1$. Let the type of N_i be

$$\rho_i \equiv \rho_{i,1} \to \cdots \to \rho_{i,m_i} \to a_i. \tag{20}$$

Then since N_i is long, types of $x_{i,1}, x_{i,2}, ...$ are exactly $\rho_{i,1}, \rho_{i,2}, ...$; i.e.

$$IAT(N_i) = \langle \rho_{i,1}, ..., \rho_{i,m_i} \rangle.$$

Just as in the proof of the lemma 17 let B_i the body of N_i for $i = 0, ..., d$. Then the type of B_i is the tail of the type N_i, namely a_i.

Define a sequence of integers $d_0, d_1, ...$ thus $d_0 = 0$ and d_{j+1} is the least $i > d_j$ such that $IAT(N_i)$ differs from all of

$$IAT(N_{d_0}), IAT(N_{d_1}), ..., IAT(N_{d_j}). \tag{21}$$

Let n be the greatest integer such that d_n is defined. The branch (18) has only d members after N_0, so $n \leq d$ and $d_n \leq d$. Then

$$0 = d_0 < d_1 < \cdots < d_n \leq d. \tag{22}$$

and for $0 \leq i \leq d$, $IAT(N_i)$ is identical to one of

$$IAT(N_{d_0}), IAT(N_{d_1}), ..., IAT(N_{d_n}). \tag{23}$$

Also the $n+1\, IAT$'s in (23) are distinct, and by the lemma 15 they are either empty or members of $NSS(\tau)$. Hence by **9E9.3(ii)**,

$$n + 1 \leq |\tau|. \tag{24}$$

Now $d_0, ..., d_n$ partition the set $\{0, 1, ..., d\}$ into the following $n+1$ non-empty sets which will be called ***IAT-intervals***:

$$\begin{array}{lcll} \mathbb{I}_j & = & \{d_j, d_j + 1, ..., d_{j+1} - 1\} & \quad (0 \leq j \leq n-1) \\ \mathbb{I}_n & = & \{d_n, d_n + 1, ..., d\}. & \end{array}$$

If $\mathbb{I}_j$ contains two numbers p, $p+r$ such that $r \geq 1$ and B_p and B_{p+r} have the same type (i.e. $a_p \equiv a_{p+r}$) we shall call $\langle p, p+r \rangle$ a ***tail-repetition***. It will be called ***minimal*** iff there is no other tail-repetition $\langle p', q' \rangle$ with

$$p \leq p' < q' \leq p + r$$

Now an $\mathbb{I}_j$ that contains no tail-repetitions must have $\leq ||\tau||$ members. Because for such an $\mathbb{I}_j$ the atoms

$$a_{d_j}, ..., a_{d_{j+i-1}}$$

must all be distinct, and by (20), each a_i occurs in ρ_i, which occurs in τ by the lemma 15, and there are only $||\tau||$ distinct atoms in τ. This argument also shows that for a minimal tail-repetition $\langle p, p+r \rangle$ we have

$$r \leq ||\tau|| \tag{25}$$

Now there are $n+1$ IAT-intervals in the given branch and $n+1 \leq ||\tau||$ by (24), so if no interval contained a tail-repetition the branch would have $\leq |\tau| \times ||\tau||$ members. But the branch has $d+1$ members and

$$d + 1 = Depth(M) + 1 \geq \mathbb{D}(\tau) + 1 > |\tau| \times ||\tau||$$

Hence at least one IAT-interval contains a tail-repetition.

We start to build M^* as follows. In the given branch take the last $\mathbb{I}_j$ containing a tail repetition, choose a minimal tail-repetition $\langle p, p+r\rangle$ in it, and change M to a new term M' by replacing B_p by B_{p+r}.

To see that M' is a genuine typed term we apply **5B2.1(ii)** similarly to the proof of the lemma 17. Using the notation of (21) in that proof, it is enough to show that the set

$$Con(B_{p+r}) \cup Con(M) \cup \Gamma_0 \cup \cdots \cup \Gamma_p \tag{26}$$

is consistent. But, as in the proof of the lemma 17, $Con(M) = 0$ and $Con(B_{p+r})$ is a subset of $\Gamma_0 \cup \cdots \cup \Gamma_{p+r}$, so (26) is consistent by (21) in the proof of the lemma 17.

It is straightforward to prove that M' is a long β-nf with the same type as M. Also that $|M'| < |M|$.

But M' might not be closed, because the change from M to M' has removed the initial abstractors of $N_{p+1}, ..., N_{p+r}$ from M, and so some free variable-occurrences in B_{p+r} that were bound in M might now be free in M'. To close M', apply the following procedure to every such variable-occurrence.

Let v be free in the occurrence of B_{p+r} in M' that has replaced B_P in M, and let v be also free in M'. Then v does not occur in a covering abstractor of this occurrence of B_{p+r} in M. But these covering abstractors are exactly the initial abstractors of $N_0, ..., N_P$ in M, so

$$v \notin IA(N_0) \cup \cdots \cup IA(N_p)$$

But M is closed, so the free v in B_{p+r} in M must be in the scope of a λv in one of $IA(N_0), ..., IA(N_{p+r})$. Hence v occurs in $IA(N_h)$ for some h with $p+1 < h < p+r$; in the notation of (19) we have

$$v \equiv x_{h,k}$$

for some $k < m_h$. And the type of v is $\rho_{h,k} \in IAT(N_h)$. Now by the definition of $d_0, ..., d_n$ and the fact that the tail-repetition $\langle p, p+r\rangle$ we are eliminating is in an interval $\mathbb{I}_j$, $IAT(N_h)$ coincides with one of

$$IAT(N_{d_0}), ..., IAT(N_{d_j});$$

say $IAT(N_h) = IAT(N_{d_q})$ for some $q \leq j$. Hence there is a variable

$$x_{d_q,k} \in IA(N_{d_q})$$

with the same type as v. Replace v by this variable throughout M'. The result will be a long β-nf with the same type and depth as M' and containing one less free variable.

Similarly replace every variable of B_{p+r}, that is free in M' by a new one which has the same type but is bound in M'. The result will be a long β-nf M" with the same type and depth as M' and which is closed.

Now M" has been obtained by removing a type-repetition from an argument-branch in M which originally contained d subarguments. And by (25) the number of subarguments removed is $< ||\tau||$. Hence

$$d - ||\tau|| \leq Depth(M") \leq d. \tag{27}$$

If $Depth(M") < d$; define $M* \equiv M"$. If not, select a branch in M" with length d and apply the above removal procedure to it, then continue shortening branches with length d until there are none left. (This process must terminate because each removal strictly reduces $|M|$.) Define $M*$ to be the first term produced by this procedure whose depth is less than d. Then

$$d - ||\tau|| \leq Depth(M*) < d$$

as required.

□

6 Conclusion

The presented search and count algorithms are efficient to showing which are the normal inhabitants of a given type τ. But other often problems isn't reached:

(i) Counting principal normal inhabitants ($Nprinc(\tau)$, $Nprinc_\eta(\tau)$) or counting principal inhabitants in general ($Princ(\tau)$). As discussed in the remark 1, there are some types τ that have the following property: given M_1^τ, M_2^τ we have that M_1 is a β-normal inhabitant but not principal, and M_2^τ is a principal but not β-normal. Characterize in general this class of types is a open problem;

(iii) Counting inhabitants in a restricted classes of terms, for example the λI-terms, BCKλ-terms or BCIλ-terms is not a trivial problem, and the proof of the crucial shrinking lemma (lemma 9) would need a reformulation for each class.

Moreover, at the first glance, the complexity of the count algorithm seems to be $\mathbb{D}(\tau) = |\tau| \times ||\tau||$, but it is not correct. Each step call for listing a set $\mathcal{A}(\tau, d)$, and this list can be large. In fact the equivalent problem that is deciding whether τ are probable in Intuitionist Implicational Logic is known as polynomial-space complete problem.