Strong Normalization of the Typed λ_{ws}-calculus

R. David[1] and B. Guillaume[2]

[1] Université de Savoie, Campus Scientifique, F-73376 Le Bourget du Lac
david@univ-savoie.fr
[2] LORIA / INRIA Lorraine, Campus Scientifique, F-54506 Vandœuvre-lès-Nancy
Bruno.Guillaume@loria.fr

Abstract. The λ_{ws}-calculus is a λ-calculus with explicit substitutions introduced in [4]. It satisfies the desired properties of such a calculus: step by step simulation of β, confluence on terms with meta-variables and preservation of the strong normalization. It was conjectured in [4] that simply typed terms of λ_{ws} are strongly normalizable. This was proved in [7] by Di Cosmo & al. by using a translation of λ_{ws} into the proof nets of linear logic. We give here a direct and elementary proof of this result. The strong normalization is also proved for terms typable with second order types (the extension of Girard's system F). This is a new result.

1 Introduction

Explicit substitutions provide an intermediate formalism which, by decomposing the β rule of the λ-calculus into more atomic steps, gives a better understanding of the execution models. The pioneer calculus with explicit substitutions, $\lambda\sigma$, was introduced by Curien & al. in [1] as a bridge between the classical λ-calculus and concrete implementations of functional programming languages. Since Melliès [6] has shown that this calculus does not preserve strong normalization, even for typed terms, finding a system satisfying the following properties became a challenge:

- step by step simulation of β,
- confluence on terms with meta-variables,
- strong normalization of the calculus of substitutions,
- preservation of strong normalization of the β-reduction.

During the last decade, various systems were presented in the literature but none of them satisfied simultaneously the previous properties. λ_{ws}, the calculus we introduced in [4], has been the first satisfying all of them. In addition to explicit substitutions, the terms of λ_{ws} are decorated with "labels". The typed version of the calculus (also introduced in [4]) shows that there is a strong link between the computational and the logical points of view: substitutions correspond to cuts and labels to weakenings. The proof that any pure λ-term which is β-strongly normalizable is still strongly normalizable in the λ_{ws}-calculus was highly technical and uses ad-hoc methods. We conjectured that the typed terms are strongly normalizable (SN). Di Cosmo, Kesner and Polonovsky [7] understood the relation between λ_{ws} and linear logic and, by using a translation of λ_{ws} into proof nets, they proved this conjecture. We give here a direct and arithmetical proof of SN for simply typed terms. This proof is based on the one for the (usual) λ-calculus due to the first author [2, 3]. We also prove, by using the standard notion of reducibility candidates, that terms typable with second order types (the extension of Girard's system F) are strongly normalizable. This result is new.

The general idea of the proofs is the following. We first give a simple characterization of strongly normalizing terms (theorem 3). This result, which is only concerned with the *untyped* calculus, is interesting by itself and may be used to prove other results on λ_{ws}. It can be seen as a kind of standardization result. Theorem 3 mainly consists of commutation results. Note that permutation of rules is also the main ingredient in the proof of [7]. Then, for $\mathcal{S}$, we use this characterization to prove, by a tricky induction, a substitution lemma (theorem 6) from which the result follows immediately. For $\mathcal{F}$, we use this characterization to prove that if a term is typed then it belongs to the interpretation of its type.

The paper is organized as follows. Section 2 gives the main notations. In section 3 we introduce some useful notions and we prove the key technical result. It is used in section 4 to prove SN for simply typed terms and in section 5 for second order types.

2 The λ_{ws}-calculus

2.1 The untyped calculus

We define here a variant of λ_{ws} which is equivalent to the one in [4]: $\langle k \rangle$ is no more primitive but becomes the abbreviation of $\langle\rangle \dots \langle\rangle$, k many times and n is coded by $\langle n \rangle 0$. Since the strong normalization of both formulations are equivalent (see proposition 1 below) and the proof is a bit simpler for the new one, we introduce here this calculus.

Definition 1. *The set of terms of λ_{ws} is defined by the following grammar:*

$$T = 0 \mid \lambda T \mid (T\ T) \mid \langle\rangle T \mid [i/T, j]T \text{ where } i, j \in \mathbb{N}.$$

and the reduction rules of the λ_{ws}-calculus are given in fig.1.

Remark 1.
- The "logical" meaning of $\langle\rangle$ and $[i/u, j]t$ is given by the typing rules. The "algorithmic" meaning is, intuitively, the following: $\langle k \rangle t$ means that each de Bruijn index in t is increased by k (as a consequence, there is no variable with de Brujin indices less than k in t) and $[i/u, j]t$ represents the term t in which the variable indexed by i is substituted by u with a re-indexing commanded by j.
- It is clear that the version of λ_{ws} presented here is a restriction of the one in [4]. For self completeness the terms and the rules of this calculus are given in the appendix. The translation ϕ from the latter to the present one is given by: $\phi(t)$ is obtained from t by replacing n by $\langle n \rangle 0$ and then $\langle k \rangle$ by $\langle\rangle \dots \langle\rangle$, k many times. In particular, $\langle 0 \rangle$ is empty.

b	$(\langle k\rangle\lambda t\, u) \longrightarrow [0/u,k]t$		
l	$[i/u,j]\lambda t \longrightarrow \lambda[i+1/u,j]t$		
a	$[i/u,j](t\, v) \longrightarrow (([i/u,j]t)\ ([i/u,j]v))$		
e_1	$[0/u,j]\langle\rangle t \longrightarrow \langle j\rangle t$		
e_2	$[i/u,j]\langle\rangle t \longrightarrow \langle\rangle[i-1/u,j]t$		$i>0$
n_1	$[i/u,j]0 \longrightarrow 0$		$i>0$
n_2	$[0/u,j]0 \longrightarrow u$		
c_1	$[i/u,j][k/v,l]t \longrightarrow [k/[i-k/u,j]v,j+l-1]t$		$k \leq i < k+l$
c_2	$[i/u,j][k/v,l]t \longrightarrow [k/[i-k/u,j]v,l][i-l+1/u,j]t$		$k+l \leq i$

Fig. 1. Reduction rules of λ_{ws}

– Note that, in this variant, the reduction rules become a bit simpler and some of them (m and n_3 in the original calculus) even disappear. Also note that rules b_1 and b_2 give a unique rule b which is in fact a *family* of rules since $\langle k\rangle$ represents a family of symbols.

Proposition 1. *If $t \to t'$ then $\phi(t) \to^+ \phi(t')$. In particular, the strong normalization of both versions of λ_{ws} are equivalent.*

Proof. Straightforward. □

2.2 The typed calculus

Definition 2. *Let $\mathcal{V}$ be a set of type variables.*

– *The set $\mathcal{S}$ of simple types is defined by: $\mathcal{S} ::= \mathcal{V} \mid \mathcal{S} \to \mathcal{S}$*
– *The set $\mathcal{F}$ of second-order types is defined by: $\mathcal{F} ::= \mathcal{V} \mid \mathcal{F} \to \mathcal{F} \mid \forall\mathcal{V}.\mathcal{F}$*

Definition 3. – *A basis Γ is an (ordered) list of types. The length of Γ is denoted by $\|\Gamma\|$.*
– *The typing rules for $\mathcal{F}$ are the given in fig.2. Note that the first element (on the left) of Γ corresponds to the variable with de Bruijn index 0. For $\mathcal{S}$, just forget $\forall_i$ and $\forall_e$.*

Proposition 2. *Both systems have subject reduction: if $\Gamma \vdash t : A$ and $t \to u$, then $\Gamma \vdash u : A$.*

Proof. We have to check that, for each rule, the typing is preserved after reduction. We give below the example of rule b. The proof is detailed in [5] for the original version of the calculus.

The typing of the b-redex $(\langle k\rangle\lambda t\ u)$ is given on the left and the typing of its reduct $[0/u,k]t$ is given on the right. We assume that $\|\Gamma\| = k$ and the last element of Γ is C.

$$\frac{}{A,\Gamma \vdash 0 : A}(Ax) \qquad \frac{\Gamma \vdash t : A}{B,\Gamma \vdash \langle\rangle t : A}(Weak)$$

$$\frac{A,\Gamma \vdash t : B}{\Gamma \vdash \lambda t : A \to B}(\to_i) \qquad \frac{\Gamma \vdash t : A \to B \quad \Gamma \vdash u : A}{\Gamma \vdash (t\,u) : B}(\to_e)$$

$$\frac{\Gamma, A, \Phi \vdash t : B \quad \Delta, \Phi \vdash u : A}{\Gamma, \Delta, \Phi \vdash [i/u, j]t : B}(Cut) \;\; \text{where } i = \|\Gamma\| \text{ and } j = \|\Delta\|$$

$$\frac{\Gamma \vdash t : A}{\Gamma \vdash t : \forall\alpha.A}(\forall_i) \text{ if } \alpha \notin \Gamma \qquad \frac{\Gamma \vdash t : \forall\alpha.A}{\Gamma \vdash t : A\{\alpha := B\}}(\forall_e)$$

Fig. 2. Typing rules of the λ_{ws}-calculus

$$\frac{\dfrac{\dfrac{\dfrac{\dfrac{A, \Delta \vdash t : B}{\Delta \vdash \lambda t : A \to B}(\to_i)}{C, \Delta \vdash \langle\rangle \lambda t : A \to B}(Weak)}{\vdots}(Weak)}{\Gamma, \Delta \vdash \langle k\rangle \lambda t : A \to B}(Weak) \qquad \Gamma, \Delta \vdash u : A}{\Gamma, \Delta \vdash (\langle k\rangle \lambda t\, u) : B}(\to_e) \qquad \frac{A, \Delta \vdash t : B \qquad \Gamma, \Delta \vdash u : A}{\Gamma, \Delta \vdash [0/u, k]t : B}(Cut)$$

□

3 Characterization of strongly normalizable terms

This section gives a characterization (Theorem 3) of strongly normalizable terms. This is the key of the proof of the strong normalization for both systems. We first need some definitions.

3.1 Some definitions

Definition 4. *The set S of substitutions and the set Σ are defined by the following grammars:*

$$S ::= \emptyset \mid [i/T, j]S \qquad \Sigma ::= \emptyset \mid \langle\rangle\Sigma \mid [i/T, j]\Sigma$$

Definition 5. *Some particular contexts are defined by the following grammars where $*$ denotes a hole and, if H is a context, $H[t]$ denotes the term obtained by replacing $*$ by t in H.*

$$C_i ::= * \mid \Sigma C_i \mid \lambda C_i \qquad C_e ::= * \mid \Sigma C_e \mid (C_e\ T) \qquad C ::= C_i[C_e]$$

Note that these contexts have a unique hole at the leftmost position. The elements of C (resp. C_i, C_e) are called head contexts (resp. i-contexts, e-contexts). Elements of T (resp. S, Σ, C) will be denoted by t, u, v, w (resp. by s, by σ, by H, K).

Notation 1 *1. We denote by $\rightarrow$ the least congruence on $T \cup C$ containing the rules of fig.1. As usual, $t \rightarrow^* t'$ (resp. $t \rightarrow^+ t'$) means that t reduces to t' by some steps (resp. at least one step) of reduction.*
2. The set of strongly normalizable terms (i.e. such that every sequence of $\rightarrow$ reductions is finite) is denoted by SN.

Lemma 1 (and notation). *Every term in T can uniquely be written as $H[0]$ or $H[(\sigma\lambda u\ v)]$ where H is an head context. The* head *of t (denoted by $\mathbf{hd}(t)$) is:*

$$\mathbf{hd}(H[0]) = H \qquad \mathbf{hd}(H[(\sigma\lambda u\ v)]) = H[(\sigma *\ v)]$$

Proof. Straightforward. □

Notation 2 *Say $t \rightarrow_{\mathbf{r}} t'$ if $t \rightarrow t'$ with the following restrictions: use only the rules a, e, c and only in $\mathbf{hd}(t)$ either at the top level or, recursively, for $[i/u, j]$ in $\mathbf{hd}(t)$, only in $\mathbf{hd}(u)$. The rule l is also permitted but only in H_i where $\mathbf{hd}(t) = H_i[H_e]$ with $H_i \in C_i$ and $H_e \in C_e$.*

Example 1.

$$[0/b, 0](\lambda[0/c, 0]1\ \ [0/d, 1]0) \rightarrow^*_{\mathbf{r}} ([0/b, 0]\lambda[0/c, 0]1\ [0/[0/b, 0]d, 1]0)$$

$$[0/a, 0]\lambda(b\ c) \rightarrow^*_{\mathbf{r}} \lambda([1/a, 0]b\ [1/a, 0]c)$$

$$[0/[0/a, 0]\lambda c, 0]b \not\rightarrow^*_{\mathbf{r}} [0/\lambda[1/a, 0]c, 0]b$$

$$(\lambda[0/a, 0]\langle\rangle b\ c) \not\rightarrow^*_{\mathbf{r}} (\lambda b\ c)$$

Lemma 2 (and notation). *The reduction $\rightarrow_{\mathbf{r}}$ is locally confluent and thus is confluent for terms such that $\mathbf{hd}(t) \in SN$. The $\mathbf{r}$-normal form of t will be denoted by $\mathbf{r}(t)$.*

Proof. Straightforward. □

Remark 2. The $\mathbf{r}$-reduction is actually strongly normalizing for every term and thus confluent. This follows immediately from the strong normalization of the calculus of substitution (i.e. all the rules except b) which is proved in [4]. We have stated the previous lemma in this way to keep this paper self contained, i.e. our proof does not need this result. Thus, in the rest of the paper, when we use $\mathbf{r}(t)$ or the confluence of $\mathbf{r}$ we have to check that $\mathbf{hd}(t) \in SN$. We will not mention this since this is always straightforward.

Definition 6. *1. Let H be an head context. Let $R(H) \subset T$, $L(H) \in C_i$ and, if $H \in C_i$, $I(H) \in T$ be defined by the following rules:*

– $R(*) = \emptyset$, $R(\lambda H) = \lambda R(H)$, $R(\sigma H) = \sigma R(H)$ *and* $R((H\ t)) = R(H) \cup \{t\}$.
– $L(*) = *$, $L(\lambda H) = \lambda L(H)$, $L(\sigma H) = \sigma L(H)$ *and* $L((H\ t)) = L(H)$.
– $I(*) = 0$, $I(H[\lambda *]) = I(H[\langle\rangle *]) = I(H)$ *and* $I(H[[i/u, j]*]) = H[\langle i\rangle u]$.

2. *An head context is* pure *if* $L(H)$ *has no substitutions.*
3. *Let* t *be a term in* T*. The set* $\mathbf{arg}(t) \subset T$ *is defined by:*
   – $\mathbf{arg}(H[0]) = R(H) \cup \{I(L(H))\}$
   – $\mathbf{arg}(H[(\sigma\lambda u\ v)]) = R(H[(*\ v)]) \cup L(H)[\sigma\lambda u]$

Remark 3. In the previous definition, the equation $R(\lambda H) = \lambda R(H)$ actually means, since $R(H)$ is a set of terms, $R(\lambda H) = \{\lambda t\ /\ t \in R(H)\}$ and similarly for $R(\sigma H) = \sigma R(H)$.

Example 2.

$$\mathbf{arg}([4/0, j](\langle 2\rangle\lambda 3\ 0)) = \{[4/0, j]\langle 2\rangle\lambda 3, [4/0, j]0\}$$

$$\mathbf{arg}([2/0, j][0/v, 2]\langle\rangle 0) = \{[2/0, j]v\}$$

Lemma 3. – *Let* $t = H[0]$*. Then* $\mathbf{r}(t)$ *can be uniquely written as* $K[s0]$ *where* K *is pure.*
– *Let* $t = H[(\sigma\lambda u\ v)]$*. Then* $\mathbf{r}(t)$ *can be uniquely written as* $K[(\langle k\rangle s\lambda u\ v_1)]$ *where* K *is pure.*

Proof. Straightforward. □

Definition 7. *Let* $s \in S$ *be a substitution, we define* $s^+ \in S$ *and* $s^\downarrow \in T$ *as follows:*

– s^+ *is defined by:* $\emptyset^+ = \emptyset$ *and* $([i/u, j]s)^+ = [i + 1/u, j]s^+$.
– $s^\downarrow$ *is defined by:* $\emptyset^\downarrow = 0$ *and* $(s[i/u, j])^\downarrow = su$ *if* $i = 0$ *and* $s^\downarrow$ *otherwise.*

Definition 8. *Let* t *be a term in* T*. The head reduct of* t *(denoted as* $\mathbf{hred}(t)$*) is defined as follows:*

– *If* $t = H[0]$ *and* $\mathbf{r}(t) = K[s0]$ *then* $\mathbf{hred}(t) = K[s^\downarrow]$.
– *If* $t = H[(\sigma\lambda u\ v)]$ *and* $\mathbf{r}(t) = K[(\langle k\rangle s\lambda u\ v_1)]$ *then* $\mathbf{hred}(t) = K[[0/v_1, k]s^+u]$.

Example 3. With terms as in the previous example, we have:

$$\mathbf{hred}([4/0, j](\langle 2\rangle\lambda 3\ 0)) = [0/[4/0, j]0, 2][3/0, j]3$$

$$\mathbf{hred}([2/0, j][0/v, 2]\langle\rangle 0) = 2$$

Theorem 3. *Let* $t \in T$ *be such that* $\mathbf{arg}(t) \subset SN$*.*

1. *Assume* $t \to^*_{\mathbf{r}} t'$ *and* $t' \in SN$*. Then* $t \in SN$*.*
2. *Assume* $\mathbf{hred}(t) \in SN$*. Then* $t \in SN$*.*

3.2 Proof of theorem 3

We first need some notations and lemmas.

Notation 4 *1. If $t \in SN$, $\eta(t)$ is the length of the longest reduction starting from t and $\eta_0(t)$ is the maximum number of b or n steps in a reduction starting from t.*

2. The complexity of a term t (denoted by $cxty(t)$) is defined by: $cxty(*) = cxty(0) = 0$, $cxty(\lambda t) = cxty(\langle\rangle t) = cxty(t) + 1$, $cxty((t\ t')) = cxty(t) + cxty(t') + 1$ and finally $cxty([i/t', j]t) = cxty(t) + cxty(t') + i + 1$.

Note that the unusual definition of $cxty([i/t', j]t)$ is due to the fact that $cxty(\langle k\rangle) = k$. It ensures that $cxty([i/u, j]) > cxty(\langle i\rangle u)$ and thus, except for $t = 0$, $cxty(u) < cxty(t)$ for any $u \in \mathbf{arg}(t)$.

Lemma 4. *Let H be an head context, u be a term and $w \in \mathbf{arg}(H[u])$. Then,*

- *either $w \in R(H)$,*
- *or $w = L(H)[v]$ for some $v \in \mathbf{arg}(u)$,*
- *or H is* not *an i-context, $u = \sigma\lambda u'$ and $w = L(H)[u]$.*

Proof. Straightforward. □

Lemma 5. *Let $H \in C$ be pure.*

1. *If $t = H[u] \in SN$ and $s \in SN$, then $H[[0/u, j]s^+0] \in SN$.*
2. *If $t = H[[0/v, k]s^+u] \in SN$, then $H[(\langle k\rangle s\lambda u\ v)] \in SN$.*

Proof. By induction on $\eta(t) + \eta(s^+0)$ for (1) and $\eta(t) + cxty(s)$ for (2). □

Lemma 6. *Let K be an head context. Assume that*

- *either $k \geq i + j$ and $w = [i/[k - i/v, l]u, j][k - j + 1/v, l]K \to_{\mathbf{r}}^* w_1 = K_1[[0/[k - i/v, l]u, j]s_1^+*]$*
- *or $i \leq k < i + j$ and $w = [i/[k - i/v, l]u, k + j - 1]K \to_{\mathbf{r}}^* w_1 = K_1[[0/[k - i/v, l]u, j]s_1^+*]$.*

Then, there is an head context K_2 such that $[i/u, j]K \to_{\mathbf{r}}^* K_2[[0/u, j_2]s_2^+*]$ and $[k/v, l]K_2[[0/u, j_2]s_2^+*] \to_{\mathbf{r}}^* w_1$.

Proof. By induction on the length of the reduction $w \to_{\mathbf{r}}^* w_1$. □

Lemma 7. *Assume $w = [i/u, j]K_1[[k/v', l]K_2] \to_{\mathbf{r}}^* w_1 = K_3[[0/u, j]s^+*]$ and $v \to v'$. Then, $[i/u, j]K_1[[k/v, l]K_2] \to_{\mathbf{r}}^* K_4[[0/u, j]s_1^+*] \to^* K_3[[0/u, j]s^+*]$ for some K_4, s_1.*

Proof. By induction on the length of the reduction $w \to_{\mathbf{r}}^* w_1$. □

Lemma 8. *1. Assume $t = H[(\sigma\lambda u\ v)] \to^* t_0 = H_0[(\langle k_0\rangle\lambda u_0\ v_0)]$. Then, there is a term $t_1 = H_1[(\langle k_1\rangle s_1\lambda u\ v_1)]$ such that $t \to_{\mathbf{r}}^* t_1 \to^* t_0$.*

2. *Assume* $t = H[0] \to^* t_0 = H_0[[0/u_0, j_0]s_0^+ 0]$. *Then,* H *can be written as* $K[[i/u,j]K_0]$ *such that* $[i/u,j]K_0 \to_{\mathbf{r}}^* K'_1 = K_1[[0/u,j]s^+ *]$ *and* $t_1 = K[K'_1][0] \to^* t_0$.

Proof. First note that we should be a bit more precise in the terms of the lemma: we implicitly assume that the potential b-redex (resp. n-redex) at the end of the left branch of t is not reduced during the reduction $t \to^* t_0$. The lemma is proved by induction on the length of the reduction $t \to^* t_0$. We give some details only for (2). They are similar and simpler for (1).

The result is clear for $t = t_0$. Assume $t \to^+ t_0$. By the induction hypothesis, $H \to H_1 = K[[i/u,j]K_0]$ for some K, u, K_0 such that $[i/u,j]K_0 \to_{\mathbf{r}}^* K'_1 = K_1[[0/u,j]s^+ *]$ and $t_1 = K[K'_1][0] \to^* t_0$. H can be written as $K_3[[i/u_1, j_1]K_2]$

- if $K_3 \to K$ or $u_1 \to u$ the result is trivial,
- if $K_2 = (* \ v)$ and $K = K_3[(* \ [i/u,j]v)]$ the result is trivial,
- if $[i/u,j]K_2 \to_{\mathbf{r}} [i/u,j]K_0$ the result is trivial,
- if $K_2 \to K_0$ but the reduction is not an $\mathbf{r}$-reduction, the result follows from lemma 7,
- if $K_3 = K[[k/v,l]*]$ and, either $[i/u,j] = [i/[k-i/v,l]u_1, j_1]$ and $K_0 = [k - j_1 + 1/v, l]K_2]$, or $[i/u,j] = [i/[k-i/v,l]u_1, l + j_1 - 1]]$ and $K_0 = K_2$, the result follows from lemma 6. □

Lemma 9. *1. Assume* $t_1 = H_1[(\sigma_1 \lambda u_1 \ v_1)] \to^* t_0 = H_0[(\langle k_0 \rangle \lambda u_0 \ v_0)]$. *Then,* $H_1[[0/v_1, k_1]s_1^+ u_1] \to^* H_0[[0/v_0, k_0]u_0]$ *where* $\mathbf{r}(\sigma_1) = \langle k_1 \rangle s_1$.

2. *Assume* $t_1 = H_1[[0/u_1, j_1]s_1^+ 0] \to^* t_0 = H_0[[0/u_0, j_0]s_0^+ 0]$. *Then,* $H_1[u_1] \to^* H_0[u_0]$.

Proof. By induction on the length of the reduction $t_1 \to^* t_0$. Look at the first reduction. Note that there is no simple relation between the original and the resulting reduction sequence and, in particular, the latter may be longer than the original. □

Lemma 10. *1. Assume* $H[(\sigma \lambda u \ v)] \to_{\mathbf{r}}^* t_0$. *Then* t_0 *has the form* $H_0[(\sigma_0 \lambda u \ v_0)]$ *and* $H[[0/v,k]s^+ u] \to_{\mathbf{r}}^* H_0[[0/v_0, k_0]s_0^+ u]$ *where* $\mathbf{r}(\sigma) = \langle k \rangle s$ *and* $\mathbf{r}(\sigma_0) = \langle k_0 \rangle s_0$.

2. *Assume* $H_0[0/u,j]s_0^+ 0 \to_{\mathbf{r}}^* t_0$. *Then* t_0 *has the form* $H_1[[0/u_1, j_1]s_1^+ 0]$ *where* $H_0 \to_{\mathbf{r}}^* H_1[s_2 *]$ *for some* s_2 *such that* $s_2[0/u,j]s_0^+ \to_{\mathbf{r}}^* [0/u_1, j_1]s_1^+$.

Proof. Straightforward. □

Lemma 11. *Let* K *be an* i*-context. Then,* $K \in SN$ *iff* $I(K) \in SN$ *and, in this case,* $\eta_0(I(K)) \leq \eta_0(K)$.

Proof. This follows immediately from the following result. Let K be an i-context, then: $K[[i/u,j]*] \in SN \Leftrightarrow K[\langle i \rangle u] \in SN$ and, in this case, $\eta_0(K[\langle i \rangle u]) \leq \eta_0(K[[i/u,j]*])$.

- ⇒ Prove, by induction on $(\eta(t), cxty(K))$ that if $t = K[s[i/u, j]*] \in SN$ then $K[d(s, i)u] \in SN$ where $d(s, i)$ is the result of moving down s through $\langle i \rangle$. It is enough to prove that, if $K[d(s, i)u] \to t'$ then $t' \in SN$. This is done by a straightforward case analysis.
- ⇐ This is proved by showing that to any sequence of reductions of $t' = K[\langle i \rangle u]$ corresponds a sequence of reductions of t with the same b or n steps. Define for $s \in S$, $\delta(s) \in \mathbb{Z}$ by: $\delta(\emptyset) = 0$ and $\delta([k/v, l]s) = \delta(s) + l - 1$.
  We show that, to a term of the form $K'[\langle i' \rangle u']$ coming from t' corresponds, for some s such that $\delta(s) < i'$, the term $K'[s[i' - \delta(s)/u', l]*]$ coming from t. This is done by a straightforward case analysis. For example, if $t' \to^* K'[[k/v, l]\langle i' \rangle u'] \to K'[\langle l + i' - 1 \rangle u']$ then $t \to^* K'[[k/v, l]s[i' - \delta(s)/u', l]*] = K'[s'[i' - \delta(s')/u', l]*]$ where $s' = [k/v, l]s$.
  It is important to note that the result on η_0 would not be true with η. This is essentially because $[k/v, l]$ can always go through $\langle i \rangle$ whereas $[k/v, l]$ cannot move down in $[i/u, j]$ if $k < i$. □

Lemma 12. *Let $t \in T$ be such that $\mathbf{arg}(t) \subset SN$ and $t \notin SN$. Then,*

1. *If $t = H[(\sigma\lambda u\ v)]$, there is a term $t_1 = H_1[(\langle k_1 \rangle s_1 \lambda u\ v_1)]$ such that $t \to_{\mathbf{r}}^* t_1$ and $H_1[[0/v_1, k_1]s_1^+ u] \notin SN$.*
2. *If $t = H[0]$ there is a term $t_1 = K[K_1[[0/u, j]s_1^+ 0]]$ such that $t \to_{\mathbf{r}}^* t_1$, t can be written as $K[[i/u, j]K_0][0]$ and $K[K_1][u] \notin SN$.*

Proof. 1. Since $\mathbf{arg}(t) \subset SN$, the potential b-redex must be reduced in an infinite reduction of t and thus such a reduction looks like: $t \to^* H_0[(\langle k_0 \rangle \lambda u_0\ v_0)] \to H_0[[0/v_0, k_0]u_0] \to ...$ and the result follows from lemmas 8 and 9.

2. Since $\mathbf{arg}(t) \subset SN$ and thus, by lemma 11, $H \in SN$, an infinite reduction of t looks like: $t \to^* H_0[[0/u_0, j_0]s_0^+ 0] \to H_0[u_0] \to ...$ and the result follows from lemma 8 and 9. □

Proof of theorem 3

1. By induction on $(\eta_0(t'), cxty(t))$. Note that the proof is by contradiction. We tried to find a constructive proof but we have been unable to find a correct one.
   - Assume first $t = H[(\sigma\lambda u\ v)]$ and $t \notin SN$. By lemma 12, let $t \to_{\mathbf{r}}^* t_0 = H_0[(\langle k_0 \rangle s_0 \lambda u\ v_0)]$ be such that $t_1 = H_0[[0/v_0, k_0]s_0^+ u] \notin SN$. By the confluence of $\to_{\mathbf{r}}^*$, let t'_0 be such that $t' \to_{\mathbf{r}}^* t'_0$ and $t_0 \to_{\mathbf{r}}^* t'_0$. By lemma 10 with the reduction $t_0 \to_{\mathbf{r}}^* t'_0$, $t'_0 = H'[(\sigma' \lambda u\ v')]$. Let $t'_1 = H'[[0/v', k']s'^+ u]$ where $\mathbf{r}(\sigma') = \langle k' \rangle s'$. Then $\eta_0(t'_1) < \eta_0(t')$ and, by lemma 10, $t_1 \to_{\mathbf{r}}^* t'_1$. It is thus enough to show that $\mathbf{arg}(t_1) \subset SN$ to get a contradiction from the induction hypothesis.
     Let $w_1 \in \mathbf{arg}(t_1)$. By lemma 4, either $w_1 \in \mathbf{arg}(t_0)$ and the result is trivial or $w_1 = L(H_0)[[0/v_0, k_0]s_0^+ w]$ for some $w \in \mathbf{arg}(u)$ or H is not an i-context and $w_1 = L(H_0)[[0/v_0, k_0]s_0^+ u]$.
     Since the second case is similar, we consider only the first one. Let $a = L(H)[(\sigma\lambda w\ v)]$ and $a' = L(H')[(\sigma' \lambda w\ v')]$. Then, $a \to_{\mathbf{r}}^* a'$ and

$\eta_0(a') \leq \eta_0(t')$ (use lemma 11 for the difficult case, i.e. when $u = K[0]$ and $w = I(L(K))$). If it is *not* the case that H is an i-context and $u = 0$, then $cxty(a) < cxty(t)$ and, by the induction hypothesis, $a \in SN$ and the result follows since $a \to^* w_1$. Otherwise, the result is trivial since it is easily seen (by induction on $(\eta(H), cxty(H))$) that, if $t = H[(\sigma\lambda 0\ v)]$ (where H is an i-context), $\mathbf{r}(\sigma) = \langle k\rangle s$ and $\mathbf{arg}(t) \subset SN$, then $H[[0/v, k]s^+0] \in SN$.

– Assume $t = H[0]$ and $t \notin SN$. By lemma 12, let $t = K[[i/u, j]K_0][0] \to^*_{\mathbf{r}} t_0 = K[H_0][[0/u, j]s_0^+0]$ be such that $t_1 = K[H_0][u] \notin SN$. By the confluence of $\to^*_{\mathbf{r}}$, let t'_0 be such that $t' \to^*_{\mathbf{r}} t'_0$ and $t_0 \to^*_{\mathbf{r}} t'_0$. By lemma 10 with the reduction $t_0 \to^*_{\mathbf{r}} t'_0$, $t'_0 = H'[[0/u', j']s'^+0]$ where $K[H_0] \to^*_{\mathbf{r}} H'[s_1*]$ for some s_1 such that $s_1[0/u, j]s_0^+ \to^*_{\mathbf{r}} [0/u', j']s'^+$. Let $t'_1 = H'[u']$. Then $\eta_0(t'_1) < \eta_0(t')$ and, by lemma 10, $t_1 \to^*_{\mathbf{r}} t'_1$. It is thus enough to show that $\mathbf{arg}(t_1) \subset SN$ to get a contradiction from the induction hypothesis.
  Let $w_1 \in \mathbf{arg}(t_1)$. By lemma 4 either $w_1 \in \mathbf{arg}(t_0)$ and the result is trivial or $w_1 = L(K[H_0])[w]$ for some $w \in \mathbf{arg}(u)$ or H is not an i-context and $w_1 = L(K[H_0])[u]$.
  Since the second case is similar, we consider only the first one. Let $a = L(K[[i/w, j]K_0])[0]$. Since $s_1 u \to^*_{\mathbf{r}} u'$, it is easy to find w' such that $s_1 w \to^*_{\mathbf{r}} w'$ and, letting $a' = L(H')[[0/w', j']s'^+0]$, $a \to^*_{\mathbf{r}} a'$ and $\eta_0(a') \leq \eta_0(t')$ (use lemma 11 for the difficult case, i.e. when $u = K[0]$ and $w = I(L(K))$). Since $cxty(a) < cxty(t)$ (except if H is an i-context and $u = 0$ but in this case again the result is trivial), by the induction hypothesis, $a \in SN$ and the result follows since $a \to^* w_1$.

2. This follows immediately from (1) and lemma 5. □

4 Strong normalization for $\mathcal{S}$

Theorem 5 below has first been proved in [7] by Di Cosmo & al. It is of course a trivial consequence of theorem 7 of section 5. However, the proof presented below is interesting in itself because it is purely arithmetical whereas the one of section 5 is not.

Theorem 5. *Typed terms of T are strongly normalizing.*

Proof. By induction on $cxty(t)$. The cases $t = 0$, $t = \lambda t'$ and $t = \langle\rangle t'$ are immediate. The case $t = [i/u, j]t'$ follows immediately from theorem 6 below. The remaining case is $t = (u\ v)$. By the induction hypothesis, u and $(0\ \langle 1\rangle v)$ are in SN. Thus, by theorem 6, $[0/u, 0](0\ \langle 1\rangle v) \in SN$ and since $[0/u, 0](0\ \langle 1\rangle v) \to^* t$ it follows that $t \in SN$. □

Theorem 6. *Assume $u, t \in T \cap SN$. Then $[i/u, j]t \in SN$.*

Proof. We prove the following. Let $u \in T \cap SN$. Then,
(1) If $t' \in T \cap SN$, then $[i/u, j]t' \in SN$.

(2) If $H \in C \cap SN$ is *pure*, then $H[u] \in SN$.
This is done by simultaneous induction on $(type(u), \eta_0(v), cxty(v), \eta_0(u))$ where $type(u)$ is the number of $\rightarrow$ in the type of u and $v = t'$ for (1) (resp. $v = H$ for (2)). The induction hypothesis will be denoted by IH.

1. $t = [i/u, j]t'$. The fact that $\mathbf{arg}(t) \subset SN$ follows immediately from the IH. By theorem 3, it is thus enough to show that $\mathbf{hred}(t) \in SN$.
   (a) If $t' = H[(\sigma\lambda v_1\ v_2)]$: since $\eta_0(\mathbf{hred}(t')) < \eta_0(t')$, it follows from the IH that $[i/u, j]\mathbf{hred}(t') \in SN$ and the result follows since $[i/u, j]\mathbf{hred}(t') \rightarrow^* \mathbf{hred}(t)$.
   (b) If $t' = H[0]$: let $\mathbf{r}(t') = K[s0]$.
   - If $s^{\downarrow} \neq 0$: since $\eta_0(\mathbf{hred}(t')) < \eta_0(t')$, it follows from the IH that $[i/u, j]\mathbf{hred}(t') \in SN$ and the result follows since $[i/u, j]\mathbf{hred}(t') \rightarrow^* \mathbf{hred}(t)$.
   - Otherwise, let $\mathbf{r}(t) = K'[s'0]$. If $s'^{\downarrow} = 0$ the result is trivial. Otherwise $s'^{\downarrow} = u'$ for some u' such that $u \rightarrow^*_{\mathbf{r}} u'$ and thus $t_1 = \mathbf{hred}(t) = K'[u']$. If K' is an i-context the result is trivial. Otherwise $K' = H'[(\langle k\rangle *\ t_0)]$. Then $t_1 = H'[(\langle k\rangle u'\ t_0)]$. It is clear that $\mathbf{arg}(t_1) \subset SN$. It is thus enough to show that $\mathbf{hred}(t_1) \in SN$.
     * If $u' = \langle k'\rangle\lambda u'_0$ and thus $\mathbf{hred}(t_1) = H'[w]$ where $w = [0/t_0, k + k']u'_0$. Since $type(t_0) < type(u)$, by the IH, $w \in SN$. By the IH, $H'[w] \in SN$ since $type(w) < type(u)$. Note that, here, we use (2).
     * Else $\mathbf{hred}(t_1) = H'[(\langle k\rangle\mathbf{hred}(u')\ t_0)] = \mathbf{hred}([i/\mathbf{hred}(u'), j]t')$. If $u' \rightarrow^+ \mathbf{hred}(u')$, the result follows from the IH. Otherwise, the result is trivial.

2. $t = H[u]$. If H is a i-context, the result is immediate. Otherwise, $H = H'[(\langle k\rangle *\ t')]$. It is clear that $\mathbf{arg}(t) \subset SN$. It remains to prove that $\mathbf{hred}(t) \in SN$.
   (a) If $u = \sigma\lambda u'$: then $\mathbf{hred}(t) = \mathbf{r}(H')[[0/t', k + k']s^+u']$ where $\mathbf{r}(\sigma) = \langle k'\rangle s$. Since $u \in SN$, $s^+u' \in SN$. By the IH since $type(t') < type(u)$, $[0/t', k + k']s^+u' \in SN$. Finally $\mathbf{hred}(t) \in SN$ since $type([0/t', k + k']s^+u') < type(u)$.
   (b) Otherwise $\mathbf{hred}(t) = H[\mathbf{hred}(u)]$. If $u \rightarrow^+ \mathbf{hred}(u)$ the result follows from the IH and otherwise the result is trivial. ☐

Remark 4. We need (2) in the proof of (1) for the following reason: we cannot always find H' and i, j such that $[i/v, 0]H'[\langle j\rangle 0] \rightarrow^* H[v]$. By choosing i large enough and j conveniently it is not difficult to get $[i/v, 0]H[\langle j\rangle 0] \rightarrow^* H[\langle j\rangle v]$ but we do not know how to get rid of $\langle j\rangle$. This is rather strange since, in the λ-calculus, this corresponds to the trivial fact that $(u\ \boldsymbol{v})$ can be written as $(x\ \boldsymbol{v})[x := u]$ where x is a fresh variable.

5 Strong normalization for $\mathcal{F}$

The proof uses the same lines as the one for the (ordinary) λ-calculus. We first define the candidates of reducibility and show some of their properties. Then, we define the interpretation of a type and we show that if t has type A then t belongs to the interpretation of A.

Definition 9. *1. If X and Y are subsets of T, $X \to Y$ denotes the set of t such that, for all $u \in X$, $(t\ u) \in Y$.*
2. The set $\mathcal{C}$ of candidates of reducibility is the smallest set which contains SN and is closed by $\to$ and intersection.
3. N_0 is the set of terms of the form $(0\ u_1...u_n)$ where $u_i \in SN$ for each i.

Lemma 13. *Assume $C \in \mathcal{C}$. Then, $N_0 \subset C \subset SN$.*

Proof. By induction on C. □

Definition 10. *An* interpretation *I is a function from $\mathcal{V}$ to $\mathcal{C}$. I is extended to $\mathcal{F}$ by: $|\alpha|_I = I(\alpha)$, $|A \to B|_I = |A|_I \to |B|_I$ and $|\forall\alpha.A|_I = \bigcap_{C \in \mathcal{C}} |A|_{I\{\alpha:=C\}}$ (where $J = I\{\alpha := C\}$ is such that $J(\alpha) = C$ and $J(\beta) = I(\beta)$ for $\beta \neq \alpha$).*

Definition 11.
- *Let $u_0, \ldots, u_{n-1}$ be a sequence of terms. We denote by $[i/\boldsymbol{u}]$ the substitution $[i/u_0, 0][i+1/u_1, 0] \ldots [i+n-1/u_{n-1}, 0]$.*
- *For $\Gamma = A_0, ..., A_{n-1}$, $\boldsymbol{u} \in |\Gamma|_I$ means that $u_i \in |A_i|_I$ for all i.*
- *A substitution s is regular if it is of the form $[i/\boldsymbol{u}]$ and $u_i \in SN$ for each i.*

Lemma 14. *Let $\boldsymbol{w}$ be a sequence of terms in SN, $s \in S$ be regular and $C \in \mathcal{C}$. Assume either $t' \to^*_{\mathbf{r}} t$ or $t' = [0/t, j]s^+0$ or $t' = (s\lambda u\ v)$ and $t = [0/v, 0]s^+u$. If $(t\ \boldsymbol{w}) \in C$, then $(t'\ \boldsymbol{w}) \in C$.*

Proof. By induction on C. The case $C = SN$ follows immediately from theorem 3. The other cases are straightforward. □

Lemma 15. *$|A\{\alpha := B\}|_I = |A|_{I\{\alpha:=|B|_I\}}$ and thus $|A|_{I\{\alpha:=B\}} = |A|_I$ if $\alpha \notin A$.*

Proof. Straightforward. □

Lemma 16. *Let I be an interpretation. Assume $\Gamma \vdash t : B$ and $\boldsymbol{u} \in |\Gamma|_I$ then $[0/\boldsymbol{u}]t \in |B|_I$.*

Proof. By induction on $\Gamma \vdash t : B$. For simplicity, we write $|A|$ instead of $|A|_I$. Assume $\boldsymbol{u} \in |\Gamma|$ and look at the last rule used in the typing derivation:

- rule Ax:

$$\frac{}{A, \Gamma \vdash 0 : A}$$

Let $v \in |A|$. By lemma 13, $v, \boldsymbol{u} \in SN$ and the result follows from lemma 14.

– rule $\to_i$:

$$\frac{A, \Gamma \vdash t : B}{\Gamma \vdash \lambda t : A \to B}$$

Let $v \in |A|$ and $w = ([0/\boldsymbol{u}]\lambda t\ v)$. By the IH, $[0/v, 0][1/\boldsymbol{u}]t \in |B|$ and the result follows from lemma 14.

– rule $\to_e$:

$$\frac{\Gamma \vdash t : A \to B \quad \Gamma \vdash v : A}{\Gamma \vdash (t\ v) : B}$$

By the IH, $[0/\boldsymbol{u}]t \in |A \to B|$ and $[0/\boldsymbol{u}]v \in |A|$. Thus $([0/\boldsymbol{u}]t\ [0/\boldsymbol{u}]v) \in |B|$ and the result follows from lemma 14.

– rule $Weak$:

$$\frac{\Gamma \vdash t : A}{B, \Gamma \vdash \langle\rangle t : A}$$

Let $v \in |B|$. By the IH $[0/\boldsymbol{u}]t \in |A|$ and the result follows from lemma 14.

– rule Cut:

$$\frac{\Gamma, A, \Phi \vdash t : B \quad \Delta, \Phi \vdash v : A}{\Gamma, \Delta, \Phi \vdash [i/v, j]t : B} \text{ where } i = \|\Gamma\| \text{ and } j = \|\Delta\|$$

Let $\boldsymbol{u}_1 \in |\Delta|$, $\boldsymbol{u}_2 \in |\Phi|$ and $w' = [0, \boldsymbol{u}][i/\boldsymbol{u}_1][i+j/\boldsymbol{u}_2][i/v, j]t$. By the IH (on the second premise), $[0/\boldsymbol{u}_1][j/\boldsymbol{u}_2]v \in |A|$. By the IH (on the first premise), $w = [0/\boldsymbol{u}][i/[0/\boldsymbol{u}_1][j/\boldsymbol{u}_2]v, 0][i+1/\boldsymbol{u}_2]t \in |B|$. Since $w' \to_{\mathbf{r}}^* w$, The result follows from lemma 14.

– rule $\forall_i$:

$$\frac{\Gamma \vdash t : A}{\Gamma \vdash t : \forall \alpha.A} \text{ if } \alpha \notin \Gamma$$

Let $C \in \mathcal{C}$. Since $\alpha \notin \Gamma$, by lemma 15, $\boldsymbol{u} \in |\Gamma|_{I\{\alpha:=C\}}$ and thus, by the IH, $[0/\boldsymbol{u}]t \in |A|_{I\{\alpha:=C\}}$. It follows that $[0/\boldsymbol{u}]t \in |\forall \alpha.A|_I$.

– rule $\forall_e$:

$$\frac{\Gamma \vdash t : \forall \alpha.A}{\Gamma \vdash t : A\{\alpha := B\}}$$

By the IH, $[0/\boldsymbol{u}]t \in |\forall \alpha.A|_I$ and thus $[0/\boldsymbol{u}]t \in |A|_{I\{\alpha:=|B|_I\}} = |A\{\alpha := B\}|_I$ (by lemma 15). □

Theorem 7. *Every typed term is strongly normalizing.*

Proof. Assume $\Gamma \vdash t : B$. By lemma 13, $\mathbf{0} \in |\Gamma|$ and thus, by lemma 16, $[0/\mathbf{0}]t \in |B|$. By lemma 13, $[0/\mathbf{0}]t \in SN$ and thus, since SN is closed by subterms, $t \in SN$. □

References

1. M. Abadi, L. Cardelli, P.-L. Curien, and J.-J. Lévy. Explicit substitutions. *Journal of Functional Programming*, 1(4):375–416, 1991.
2. R. David. A short proof of the strong normalization of the simply typed λ-calculus. available: www.lama.univ-savoie.fr/~david.

3. R. David. Normalization without reducibility. *Annals of Pure and Applied Logic*, 107:121–130, 2001.
4. R. David and B. Guillaume. A λ-calculus with explicit weakening and explicit substitution. *Mathematical Structures for Computer Science*, 11:169–206, 2001.
5. B. Guillaume. *Un calcul de substitutions avec étiquettes.* Phd thesis, Université de Savoie, 1999. available: http://www.loria.fr/~guillaum/publications/Gui99.ps.gz.
6. P.-A. Melliès. Typed λ-calculi with explicit substitutions may not terminate. *Proceedings of Typed Lambda Calculi and Applications 95* in *Lecture Notes in Computer Science*, 902:328–334, 1995.
7. D. Kesner R. Di Cosmo and E. Polonovsky. Proof nets and explicit subsitutions. In *Fossacs'2000 and LNCS, 1784 : 63-81*, 2000.

6 Appendix

The set of terms and the reduction rules of the original calculus of [4] are:

Terms

$$T = \underline{n} \mid \lambda T \mid (T\ T) \mid \langle k \rangle T \mid [i/T, j]T \text{ where } n, k, i, j \in \mathbb{N}.$$

Rules

b_1	$(\lambda t\ u) \longrightarrow [0/u, 0]t$		
b_2	$(\langle k \rangle \lambda t\ u) \longrightarrow [0/u, k]t$		
l	$[i/u, j]\lambda t \longrightarrow \lambda[i+1/u, j]t$		
a	$[i/u, j](t\ v) \longrightarrow (([i/u, j]t)\ ([i/u, j]v))$		
e_1	$[i/u, j]\langle k \rangle t \longrightarrow \langle j+k-1 \rangle t$	$i < k$
e_2	$[i/u, j]\langle k \rangle t \longrightarrow \langle k \rangle [i-1/u, j]t$	$k \leq i$
n_1	$[i/u, j]\underline{n} \longrightarrow \underline{n}$	$n < i$
n_2	$[i/u, j]\underline{n} \longrightarrow \langle i \rangle u$	$n = i$
n_3	$[i/u, j]\underline{n} \longrightarrow \underline{n+j-1}$	$i < n$
c_1	$[i/u, j][k/v, l]t \longrightarrow [k/[i-k/u, j]v, j+l-1]t$	$k \leq i < k+l$
c_2	$[i/u, j][k/v, l]t \longrightarrow [k/[i-k/u, j]v, l][i-l+1/u, j]t$	$k+l \leq i$
m	$\langle i \rangle \langle j \rangle t \longrightarrow \langle i+j \rangle t$	