Intersection type systems and explicit substitutions calculi

Daniel Lima Ventura[1⋆], Mauricio Ayala-Rincón[1⋆⋆], and Fairouz Kamareddine[2]

[1] Grupo de Teoria da Computação, Dep. de Matemática Universidade de Brasília, Brasília D.F., Brasil

[2] School of Mathematical and Computer Sciences Heriot-Watt University, Edinburgh, Scotland UK

{ventura,ayala}@mat.unb.br, fairouz@macs.hw.ac.uk

Abstract. The λ-calculus with de Bruijn indices, called λ_{dB}, assembles each α-class of λ-terms into a unique term, using indices instead of variable names. Intersection types provide finitary type polymorphism satisfying important properties like principal typing, which allows the type system to include features such as data abstraction (modularity) and separate compilation. To be closer to computation and to simplify the formalisation of the atomic operations involved in β-contractions, several explicit substitution calculi were developed most of which are written with de Bruijn indices. Although untyped and simply types versions of explicit substitution calculi are well investigated, versions with more elaborate type systems (e.g., with intersection types) are not. In previous work, we presented a version for λ_{dB} of an intersection type system originally introduced to characterise principal typings for β-normal forms and provided the characterisation for this version. In this work we introduce intersection type systems for two explicit substitution calculi: the $\lambda\sigma$ and the λs_e. These type system are based on a type system for λ_{dB} and satisfy the basic property of subject reduction, which guarantees the preservation of types during computations.

1 Introduction

The λ-calculus à la de Bruijn [deBruijn72], λ_{dB} for short, was introduced by the Dutch mathematician N.G. de Bruijn in the context of the project Automath [NGdV94] and has been adopted for several calculi of explicit substitutions ever since, e.g. [deBruijn78,ACCL91,KR97]). Term variables are represented by indices instead of names in λ_{dB}, assembling each α-class of terms in the λ-calculus [Barendregt84] into a unique term with de Bruijn indices, thus making it more *"machine-friendly"* than its counterparts. The $\lambda\sigma$- [ACCL91] and the λs_e- [KR97] calculi have applications in higher order unification, HOU for short [DHK2000,AK01]. These explicit substitution calculi with de Bruijn

⋆ Supported by a PhD scholarship at the Universidade de Brasília.

⋆⋆ Supported by the Fundação de Apoio à Pesquisa do Distrito Federal [FAPDF 8-004/2007]

indices have been investigated for both type free and simply typed versions but to the best of our knowledge there is no work on more elaborate type systems such as intersection types.

Intersection types, IT for short, were introduced as an extension to simple types, in order to provide a characterisation of strongly normalising λ-terms [CDC78,CDC80,Pottinger80]. In programming, the IT discipline is of interest because λ-terms corresponding to correct programs not typable in the standard Curry type assignment system [CF58], or in some polymorphic extensions as the one present in ML [Milner78], are typable with IT. Moreover, some IT systems satisfy the principal typing property, PT for short, which means that for any typable term M there is a type judgement $\Gamma \vdash M : \tau$ representing all possible typings $\langle \Gamma' \vdash \tau' \rangle$ of M in the corresponding type system. Principal typings has been studied for some IT systems [CDV80,RV84,Rocca88,Bakel95,KW04] and in [CDV80,RV84] it was shown that for a term M, the principal typing of M's β-normal form, β-nf for short, is principal for M itself.

In [VAK09] we introduced an IT system for the λ_{dB}, based on the type system given in [KN07], and proved it to satisfy the subject reduction property, SR for short, which states that types under β-reduction are preserved: whenever $\Gamma \vdash M \ : \ \sigma$ and M β-reduces into N, then $\Gamma \vdash N \ : \ \sigma$. Due to the interaction between sequential type contexts and the subtyping relation, the system in [VAK09] is not relevant in the sense of [DG94], whereas the system of [KN07] is. Hence, in [VAK10] we introduce a relevant IT system for λ_{dB}. This system is a de Bruijn version of the system originally introduced in [SM96a], for which we established a characterisation of the syntactic structure of PT for β-nfs.

In this paper we concentrate on the SR property, with a discussion originally presented in [VAK10], and we prove for the first time the property for the β-contraction in λ_{dB} with some considerations. We then propose a variant for the type system, which is a de Bruijn version of the system in [SM97]. We also give the first IT systems for $\lambda\sigma$ and λs_e, which we base on this variant, and we establish that they both have SR for the full rewriting system. As a preliminary step to obtain the system for λs_e, we introduce an IT system for λs [KR95], based on the system of [VAK10], with similar properties such as relevance and SR for the simulation of β-contraction.

Below, we present the untyped versions of the λ_{dB}, λs, λs_e and $\lambda\sigma$ calculi. Section 2 consists of two parts. In Subsection 2.1 we present the IT systems λ_{dB}^{SM} and $\lambda_{dB}^{SM_r}$, followed by the relevance property and a discussion of the SR property. In Subsection 2.2 we present the new work on the system λ_{dB}^{SM}, introducing some properties related to the proof of SR for β-contraction in the type system, which is discussed at the end of the second part. In Section 3 we introduce the IT system λs^{SM} for λs and the system $\lambda {s_e}^{\wedge}$ for λs_e, with their respective properties. In Section 4, the IT system for $\lambda\sigma$ is introduced followed by its properties.

1.1 λ-calculus with de Bruijn indices

Definition 1 (Set Λ_{dB}). *The set of λ_{dB}-term, denoted by Λ_{dB}, is inductively defined for $n \in \mathbb{N}^* = \mathbb{N} \setminus \{0\}$ by:* $M, N \in \Lambda_{dB} ::= \underline{n} \,|\, (M\ N) \,|\, \lambda.M$.

The index $\underline{i}$ is bound if it is inside i λ's and otherwise it is free. We introduce the following subsets in order to present a formal definition of the set of free indices for some term.

Definition 2. *Let $N \subset \mathbb{N}^*$ and $k \geq 0$. We define:*
1. $N \backslash k = \{n - k \,|\, n \in N\}$ *3.* $N + k = \{n + k \,|\, n \in N\}$
2. $N_{>k} = \{n \in N \,|\, n > k\}$ *4.* $N_{\leq k} = \{n \in N \,|\, n \leq k\}$, $N_{<k} = \{n \in N \,|\, n < k\}$

Definition 3. *$FI(M)$, the* **set of free indices** *of $M \in \Lambda_{dB}$, is defined by:*
$FI(\underline{n}) = \{\underline{n}\}$ $FI(M_1\ M_2) = FI(M_1) \cup FI(M_2)$ $FI(\lambda.M) = FI(M) \backslash 1$

The free indices correspond to the notion of free variables in the λ-calculus with names, hence M is called closed when $FI(M) \equiv \emptyset$. The greatest value of $FI(M)$ is denoted by $sup(M)$. In [VAK09] we give the formal definitions of those concepts. Terms like $((\dots((M_1\ M_2)\ M_3)\dots)\ M_n)$ are written $(M_1\, M_2\, \cdots\, M_n)$, as usual. The β-contraction definition in this notation needs a mechanism which detects and updates free indices of terms. Intuitively, the **lift** of M, denoted by M^+, corresponds to an increment by 1 of all free indices occurring in M. Thus, we are able to present the definition of the substitution used by β-contractions, similarly to the one presented in [AK01].

Definition 4. *Let $m, n \in \mathbb{N}^*$. The* ***β-substitution*** *for free occurrences of $\underline{n}$ in $M \in \Lambda_{dB}$ by term N, denoted as $\{\underline{n}/N\}M$, is defined inductively by*

1. $\{\underline{n}/N\}(M_1\ M_2) = (\{\underline{n}/N\}M_1\ \{\underline{n}/N\}M_2)$ *3.* $\{\underline{n}/N\}\underline{m} = \begin{cases} \underline{m-1}, & \text{if } m > n \\ N, & \text{if } m = n \\ \underline{m}, & \text{if } m < n \end{cases}$

2. $\{\underline{n}/N\}(\lambda.M_1) = \lambda.\{\underline{n+1}/N^+\}M_1$

Observe that in item 2 of Definition 4, the lift operator is used to avoid the capture of free indices in N. We define β-contraction as usual (e.g. see [AK01]).

Definition 5. ***β-contraction*** *in λ_{dB} is defined by $(\lambda.M\ N) \rightarrow_\beta \{\underline{1}/N\}M$.*

Notice that item 3 in Definition 4 is the mechanism which does the substitution and updating of the free indices in M as a consequence of the elimination of the lead abstractor. ***β*-reduction** is defined to be the λ-compatible closure of β-contraction defined above. A term is in ***β*-normal form**, β-nf for short, if there is no β-reduction to be done.

When $\underline{i} \notin FI(M)$, then we have that $\{\underline{i}/N\}M = M^{-i}$, where M^{-i} is the term M in which indices greater than i are decreased by one. We call this an **empty substitution** because no index is replaced by an instance of term N. The β-contraction $(\lambda.M\ N) \rightarrow \{\underline{1}/N\}M$ is thus called an **empty application**.

1.2 The λs_e-calculus

The λs-calculus is a proper extension of the λ_{dB}-calculus. Two operators σ and φ are introduced for substitution and updating, respectively, to control the atomisation of the substitution operation by arithmetic constraints.

Definition 6 (Set Λs). *The set of λs-terms, denoted by Λs, is inductively defined for $n, i, j \in \mathbb{N}^*$ and $k \in \mathbb{N}$ by:* $M, N \in \Lambda s ::= \underline{n} \,|\, (M\ N) \,|\, \lambda.M \,|\, M\sigma^i N \,|\, \varphi^j_k M$

The term $M\sigma^i N$ represents the procedure to obtain the term $\{\underline{i}/N^{+^{(i-1)}}\}M$; i.e., the substitution of the free occurrences of $\underline{i}$ in M by N with its free indices incremented by $(i-1)$, updating the free indices on both terms. The term $\varphi^j_k M$ represents $j-1$ applications of the k-lift to the term M; i.e., $M^{+k^{(j-1)}}$. Table 1 contains the rewriting rules of the λs_e-calculus as given in [KR97]. The bottom six rules of Table 1 are those which extend the λs-calculus [KR95] to λs_e [KR97]. They ensure the confluence of the λs_e-calculus on open terms and its application to the HOU problem [AK01]. In this paper we work with the same set Λs of terms for both calculi.

$(\lambda.M\ N)$	$\longrightarrow M\,\sigma^1 N$	(σ-generation)
$(\lambda.M)\sigma^i N$	$\longrightarrow \lambda.(M\sigma^{i+1}N)$	(σ-λ-transition)
$(M_1\ M_2)\sigma^i N$	$\longrightarrow ((M_1\sigma^i N)\ (M_2\sigma^i N))$	(σ-app-trans.)
$\underline{n}\,\sigma^i N$	$\longrightarrow \begin{cases} \underline{n-1} & \text{if } n > i \\ \varphi^i_0 N & \text{if } n = i \\ \underline{n} & \text{if } n < i \end{cases}$	(σ-destruction)
$\varphi^i_k(\lambda.M)$	$\longrightarrow \lambda.(\varphi^i_{k+1}M)$	(φ-λ-trans.)
$\varphi^i_k(M_1\ M_2)$	$\longrightarrow ((\varphi^i_k M_1)\ (\varphi^i_k M_2))$	(φ-app-trans.)
$\varphi^i_k\,\underline{n}$	$\longrightarrow \begin{cases} \underline{n+i-1} & \text{if } n > k \\ \underline{n} & \text{if } n \le k \end{cases}$	(φ-destruction)
$(M_1\sigma^i M_2)\sigma^j N$	$\longrightarrow (M_1\sigma^{j+1}N)\sigma^i(M_2\sigma^{j-i+1}N)$ if $i \le j$	(σ-σ-trans.)
$(\varphi^i_k M)\sigma^j N$	$\longrightarrow \varphi^{i-1}_k M$ if $k < j < k+i$	(σ-φ-trans. 1)
$(\varphi^i_k M)\sigma^j N$	$\longrightarrow \varphi^i_k(M\sigma^{j-i+1}N)$ if $k+i \le j$	(σ-φ-trans. 2)
$\varphi^i_k(M\sigma^j N)$	$\longrightarrow (\varphi^i_{k+1}M)\sigma^j(\varphi^i_{k+1-j}N)$ if $j \le k+1$	(φ-σ-trans.)
$\varphi^i_k(\varphi^j_l M)$	$\longrightarrow \varphi^j_l(\varphi^i_{k+1-j}M)$ if $l+j \le k$	(φ-φ-trans. 1)
$\varphi^i_k(\varphi^j_l M)$	$\longrightarrow \varphi^{j+i-1}_l M$ if $l \le k < l+j$	(φ-φ-trans. 2)

Table 1. The rewriting system of the λs_e-calculus

$=_{s_e}$ denotes the equality for the associate substitution calculus, denoted as s_e, induced by all the rules except (σ-generation). The rewriting system obtained by removing from s_e the bottom six rules presented in Table 1 is called the s-calculus, which is the substitution calculus associated with λs. In order to have a syntactic characterisation related to empty applications and substitutions, as the free indices for λ_{dB}, we present the available indices, a notion analogous to that of available variables introduced in [LLDDvB04].

Definition 7. *$AI(M)$, the* **set of available indices** *of $M \in \Lambda s$ is defined by:*
$AI(\underline{n}) = \{\underline{n}\}$ $AI(\lambda.M) = AI(M)\backslash 1$ $AI(M_1\ M_2) = AI(M_1) \cup AI(M_2)$ *and*
$AI(\varphi^i_k M) = AI(M)_{\le k} \cup (AI(M)_{>k} + (i-1))$

$$AI(M\sigma^i N) = \begin{cases} AI(M^{-i}) \cup AI(\varphi^i_0 N), & \textit{if } i \in AI(M) \\ AI(M^{-i}), & \textit{if } i \notin AI(M) \end{cases}$$

where $AI(M^{-i})$ denotes $AI(M)_{<i} \cup (AI(M)_{>i})\backslash 1$.

The greatest value of $AI(M)$ is denoted by $sav(M)$.

1.3 The $\lambda\sigma$-calculus

The $\lambda\sigma$-calculus is given by a first-order rewriting system, which makes substitutions explicit by extending the language with two sorts of objects: **terms** and **substitutions** which are called $\lambda\sigma$-expressions.

Definition 8 (Set $\Lambda\sigma$). *The set of $\lambda\sigma$-expressions, denoted by $\Lambda\sigma$, is formed by the set $\Lambda\sigma^t$ of terms and the set $\Lambda\sigma^s$ of substitutions, inductively defined by:*

$$M, N \in \Lambda\sigma^t ::= \underline{1} \,|\, (M\ N) \,|\, \lambda.M \,|\, M[S] \qquad S \in \Lambda\sigma^s ::= id \,|\, \uparrow \,|\, M.S \,|\, S \circ S$$

Substitutions can intuitively be thought of as lists of the form $N/\underline{i}$ indicating that the index $\underline{i}$ ought to be replaced by the term N. The expression id represents a substitution of the form $\{\underline{1}/\underline{1}, \underline{2}/\underline{2}, \ldots\}$ whereas $\uparrow$ is the substitution $\{\underline{i{+}1}/\underline{i} \,|\, i{\in}\mathbb{N}^*\}$. The expression $S{\circ}S$ represents the composition of substitutions. Moreover, $\underline{1}\,[\uparrow^n]$, where $n \in \mathbb{N}^*$, codifies the de Bruijn index $\underline{n{+}1}$ and $\underline{i}\,[S]$ represents the value of $\underline{i}$ through the substitution S, which can be seen as a function $S(i)$. The substitution $M.S$ has the form $\{M/\underline{1}, S(i)/\underline{i+1}\}$ and is called the **cons of** M **in** S. $M[N.id]$ starts the simulation of the β-reduction of $(\lambda.M\ N)$ in $\lambda\sigma$. Thus, in addition to the substitution of the free occurrences of the index $\underline{1}$ by the corresponding term, free occurrences of indices should be decremented because of the elimination of the abstractor. Table 2 lists the rewriting system of the $\lambda\sigma$-calculus, as presented in [DHK2000], without the (Eta) rule.

$(\lambda.M\ N)$	$\longrightarrow M[N.id]$	$(Beta)$	$(\lambda.M)[S]$	$\longrightarrow \lambda.(M[1.(S\circ\uparrow)])$	(Abs)
$(M\ N)[S]$	$\longrightarrow (M[S]\ N[S])$	(App)	$\uparrow\circ(M.S)$	$\longrightarrow S$	$(ShiftCons)$
$M[id]$	$\longrightarrow M$	(Id)	$(S_1\circ S_2)\circ S_3$	$\longrightarrow S_1\circ(S_2\circ S_3)$	$(AssEnv)$
$1[S].(\uparrow\circ S)$	$\longrightarrow S$	$(Scons)$	$(M.S)\circ T$	$\longrightarrow M[T].(S\circ T)$	$(MapEnv)$
$(M[S])[T]$	$\longrightarrow M[S\circ T]$	$(Clos)$	$1.\uparrow$	$\longrightarrow id$	$(VarShift)$
$id\circ S$	$\longrightarrow S$	(IdL)	$1[M.S]$	$\longrightarrow M$	$(VarCons)$
$S\circ id$	$\longrightarrow S$	(IdR)			

Table 2. The rewriting system for the $\lambda\sigma$-calculus

This system is equivalent to that of [ACCL91]. The associated substitution calculus, denoted by σ, is the one induced by all the rules except (Beta), and its equality is denoted as $=_\sigma$.

2 Intersection type systems for the λ_{dB}-calculus

The intersection type systems presented in this paper have the same set of types $\mathcal{T}$, of the so called restricted intersection types. The intersection types in $\mathcal{T}$ do not occur immediately on the right of an $\rightarrow$. Besides that, the intersection is linear thus non idempotent. The type contexts in type systems with de Bruijn indices are sequences of types. Below, we present the definitions of these concepts.

Definition 9. *1. Let $\mathcal{A}$ be a denumerably infinite* **set of type variables** *and let α, β range over $\mathcal{A}$.*

$$\frac{}{\underline{1}:\langle\tau.nil\vdash\tau\rangle}\text{ var}\qquad\frac{\underline{n}:\langle\Gamma\vdash\tau\rangle}{\underline{n+1}:\langle\omega.\Gamma\vdash\tau\rangle}\text{ varn}\qquad\frac{M:\langle u.\Gamma\vdash\tau\rangle}{\lambda.M:\langle\Gamma\vdash u\to\tau\rangle}\to_i$$

$$\frac{M_1:\langle\Gamma\vdash\omega\to\tau\rangle\qquad M_2:\langle\Delta\vdash\sigma\rangle}{(M_1\;M_2):\langle\Gamma\wedge\Delta\vdash\tau\rangle}\to'_e\qquad\frac{M:\langle nil\vdash\tau\rangle}{\lambda.M:\langle nil\vdash\omega\to\tau\rangle}\to'_i$$

$$\frac{M_1:\langle\Gamma\vdash\wedge_{i=1}^n\sigma_i\to\tau\rangle\qquad M_2:\langle\Delta^1\vdash\sigma_1\rangle\;\ldots\;M_2:\langle\Delta^n\vdash\sigma_n\rangle}{(M_1\;M_2):\langle\Gamma\wedge\Delta^1\wedge\cdots\wedge\Delta^n\vdash\tau\rangle}\to_e$$

Fig. 1. Typing rules of system λ_{dB}^{SM}

2. *The set* $\mathcal{T}$ *of* **restricted intersection types** *is defined by:*
   $\tau,\sigma\in\mathcal{T}::=\mathcal{A}\,|\,\mathcal{U}\to\mathcal{T}\qquad u\in\mathcal{U}::=\omega\,|\,\mathcal{U}\wedge\mathcal{U}\,|\,\mathcal{T}$
   Types are quotiented by taking $\wedge$ *to be commutative, associative and to have* ω *as the neutral element.*
3. **Contexts** *are ordered lists of* $u\in\mathcal{U}$*, defined by:* $\Gamma::=nil\,|\,u.\Gamma$*.* Γ_i *denotes the* i*-th element of* Γ *and* $|\Gamma|$ *denotes the length of* Γ*. We let* $\omega^{\underline{n}}$ *denote the sequence* $\omega.\omega.\cdots.\omega$ *of length* n*, called* **omega context***, and let* $\omega^{\underline{0}}.\Gamma=\Gamma$*. The extension of* $\wedge$ *to contexts is done by taking nil as the neutral element and* $(u_1.\Gamma)\wedge(u_2.\Delta)=(u_1\wedge u_2).(\Gamma\wedge\Delta)$*. Hence,* $\wedge$ *is commutative and associative on contexts.*
4. *Let* $u'\sqsubseteq u$ *if there exists* v *such that* $u=u'\wedge v$ *and* $u'\sqsubset u$ *if* $v\neq\omega$*. Let* $\Gamma'\sqsubseteq\Gamma$ *if there exists* Δ *such that* $\Gamma=\Gamma'\wedge\Delta$*, where neither* Γ' *nor* Δ *are omega contexts and* $\Gamma'\sqsubset\Gamma$ *if* $\Delta\neq nil$*.*

The set $\mathcal{T}$ defined here is equivalent to the one defined in [SM96a]. Type judgements will be of the form $M:\langle\Gamma\vdash_{\mathfrak{S}}\tau\rangle$, meaning that in system $\mathfrak{S}$, term M has type τ in context Γ (where $FI(M)$ are handled). Briefly, M has type τ with Γ in $\mathfrak{S}$ or $\langle\Gamma\vdash\tau\rangle$ is a typing of M in S. The $\mathfrak{S}$ is omitted whenever its is clear which system is being referred to.

2.1 The system λ_{dB}^{SM}

We present in this section the systems λ_{dB}^{SM} and $\lambda_{dB}^{SM_r}$, introduced in [VAK10]. The system $\lambda_{dB}^{SM_r}$ is the de Bruijn version of the system presented in [SM96a], used to characterise principal typings (PT) for β-nfs.

Definition 10. *1. The typing rules for system* λ_{dB}^{SM} *are given in Figure 1.*
2. *System* $\lambda_{dB}^{SM_r}$ *is obtained from system* λ_{dB}^{SM}*, by replacing the rule* var *by rule* var$_r$*:* $\dfrac{}{\underline{1}:\langle\sigma_1\to\cdots\to\sigma_n\to\alpha.nil\vdash\sigma_1\to\cdots\to\sigma_n\to\alpha\rangle}\;(n\geq 0)$.

Proposition 1. λ_{dB}^{SM} *is a proper extension of* $\lambda_{dB}^{SM_r}$*.*

Hence, the properties stated for the system λ_{dB}^{SM} are also true for the system $\lambda_{dB}^{SM_r}$. The following lemma states that λ_{dB}^{SM} is relevant in the sense of [DG94].

Lemma 1 (Relevance for λ_{dB}^{SM} [VAK10]). *If* $M:\langle\Gamma\vdash_{\lambda_{dB}^{SM}}\tau\rangle$*, then* $|\Gamma|=sup(M)$ *and* $\forall 1\leq i\leq|\Gamma|$*,* $\Gamma_i\neq\omega$ *iff* $\underline{i}\in FI(M)$*.*

Note that, by Lemma 1 above, system λ_{dB}^{SM} is not only relevant but there is a strict relation between the free indices of terms and the length of contexts in their typings. In [VAK10] we give a characterisation of PT for β-nfs in $\lambda_{dB}^{SM_r}$.

Despite the fact that all β-nfs are typable in $\lambda_{dB}^{SM_r}$, the subject reduction property fails for both $\lambda_{dB}^{SM_r}$ and λ_{dB}^{SM}. In the following, we will give counterexamples to show that neither subject expansion nor reduction holds.

Example 1. In order to have the subject expansion property, we need to prove the statement: If $\{\underline{1}/N\}M:\langle\Gamma\vdash\tau\rangle$ then $(\lambda.M\ N):\langle\Gamma\vdash\tau\rangle$. Let $M\equiv\lambda.\underline{1}$ and $N\equiv\underline{3}$, hence $\{\underline{1}/\underline{3}\}\lambda.\underline{1}=\lambda.\underline{1}$. We have that $\lambda.\underline{1}:\langle nil\vdash\alpha\rightarrow\alpha\rangle$. Thus, $\lambda.\lambda.\underline{1}:\langle nil\vdash\omega\rightarrow\alpha\rightarrow\alpha\rangle$ and $\underline{3}:\langle\omega.\omega.\beta.nil\vdash\beta\rangle$, then $(\lambda.\lambda.\underline{1}\ \underline{3}):\langle\omega.\omega.\beta.nil\vdash\alpha\rightarrow\alpha\rangle$.

For subject reduction, we need the statement: If $(\lambda.M\ N):\langle\Gamma\vdash\tau\rangle$ then $\{\underline{1}/N\}M:\langle\Gamma\vdash\tau\rangle$. Note that if we take M and N as in the example above, we get the same problem as before but the other way round. In other words, we have a restriction on the original context after the β-reduction, since we loose the typing information regarding $N\equiv\underline{3}$. □

One possible solution is to replace rule $\rightarrow'_e$ by: $\dfrac{M:\langle\Gamma\vdash\omega\rightarrow\tau\rangle}{(M\ N):\langle\Gamma\vdash\tau\rangle}$.

This approach was originally presented in [SM96b]. However, the type system obtained there does not have the property described in Lemma 1 since we would not have the typing information for all the free indices occurring in a term. We present a lemma at the end of the present section, stating the property related to relevance for this variant.

The other way to try to achieve the desired properties is to think about the meaning of the properties themselves. Since, by Lemma 1, the system is related to relevant logic (cf. [DG94]), the notion of expansion and restriction of contexts is an interesting way to talk about subject expansion and reduction. These concepts were presented in [KN07] for environments. We introduce the notion of restriction for sequential contexts in Subsection 2.2. This approach of restriction/expansion for contexts is not sufficient to have the subject expansion property because the rule $\rightarrow'_e$ has the typability of the argument as a premiss. Hence, for any non typable term N, $\{\underline{1}/N\}\underline{2}$ is typable while $(\lambda.\underline{2}\ N)$ is not typable in system λ_{dB}^{SM}. Below, we define the system which is the basis for the IT systems we propose for λs_e and $\lambda\sigma$.

Definition 11 (The system $\lambda_{dB}^{\wedge}$). *The system $\lambda_{dB}^{\wedge}$ is obtained from system λ_{dB}^{SM}, replacing the rule $\rightarrow'_e$ by the following rule:* $\dfrac{M:\langle\Gamma\vdash\omega\rightarrow\tau\rangle}{(M\ N):\langle\Gamma\vdash\tau\rangle}\ \rightarrow_e^{\omega}$.

The following property is related to relevance in this system.

Lemma 2. *If $M:\langle\Gamma\vdash_{\lambda_{dB}^{\wedge}}\tau\rangle$ and $|\Gamma|=m>0$ then $\Gamma_m\neq\omega$ and $\forall 1\leq i\leq|\Gamma|$, $\Gamma_i\neq\omega$ implies $\underline{i}\in FI(M)$.*

Proof. By induction on the derivation $M:\langle\Gamma\vdash_{\lambda_{dB}^{\wedge}}\tau\rangle$.

2.2 Subject reduction for system λ_{dB}^{sM}

We present here the properties of system λ_{dB}^{sM} used in the proof of SR, presented at the end of this part. The generation lemmas for λ_{dB}^{sM} were presented in [VAK10] and we omit them here due to lack of space. Below, we give a lemma which relates typings and the updating operator.

Lemma 3 (Updating). *Let $M:\langle \Gamma \vdash_{\lambda_{dB}^{sM}} \tau\rangle$. If $i \geq |\Gamma|$ then $M^{+i}:\langle \Gamma \vdash_{\lambda_{dB}^{sM}} \tau\rangle$. Otherwise, if $0 \leq i < |\Gamma|$ then $M^{+i}:\langle \Gamma_{\leq i}.\omega.\Gamma_{>i} \vdash_{\lambda_{dB}^{sM}} \tau\rangle$.*

Observe that when $i \geq |\Gamma|$ then by the relevance of system λ_{dB}^{sM} we have that $i \geq sup(M)$ thus $M^{+i} = M$ (cf. [VAK09]). Otherwise, the free indices of M greater then i are incremented by one, then we need to add the ω at the $(i+1)$-th position on the sequential context to guarantee the typability for term M^{+i}. We now can introduce the substitutions lemmas.

Lemma 4 (Substitution). *Let $M:\langle \Gamma \vdash_{\lambda_{dB}^{sM}} \tau\rangle$.*

1. *If $i > |\Gamma|$ then, for any $N \in \Lambda_{dB}$, $\{\underline{i}/N\}M:\langle \Gamma \vdash_{\lambda_{dB}^{sM}} \tau\rangle$.*
2. *If $\Gamma_i = \omega$ where $0 < i < |\Gamma|$, then $\{\underline{i}/N\}M:\langle \Gamma_{<i}.\Gamma_{>i} \vdash_{\lambda_{dB}^{sM}} \tau\rangle$.*
3. *Let $\Gamma_i = \wedge_{j=1}^{m}\sigma_j$, where $0 < i \leq |\Gamma|$, and $\forall 1 \leq j \leq m$, $N : \langle nil \vdash_{\lambda_{dB}^{sM}} \sigma_j\rangle$. If $sup(M) = i$ then $\{\underline{i}/N\}M : \langle \Gamma_{<k}.nil \vdash_{\lambda_{dB}^{sM}} \tau\rangle$ for $k = sup(\{\underline{i}/N\}M)$. Otherwise, $\{\underline{i}/N\}M:\langle \Gamma_{<i}.\Gamma_{>i} \vdash_{\lambda_{dB}^{sM}} \tau\rangle$.*
4. *Let $\Gamma_i = \wedge_{j=1}^{m}\sigma_j$, where $0 < i \leq |\Gamma|$, and $N \in \Lambda_{dB}$ s.t. $sup(N) \geq i$. If $\forall 1 \leq j \leq m$, $N:\langle \Delta^j \vdash_{\lambda_{dB}^{sM}} \sigma_j\rangle$ then $\{\underline{i}/N\}M:\langle (\Gamma_{<i}.\Gamma_{>i}) \wedge \Delta^1 \wedge \cdots \wedge \Delta^m \vdash_{\lambda_{dB}^{sM}} \tau\rangle$.*

Hence, we have the relation between M and N typings and the typing for term $\{\underline{i}/N\}M$. Note that, whenever N is typable, items 1 and 2 represent the loss of its type information. Therefore, we need the restriction property for sequential contexts, introduced below, to establish the SR property.

Definition 12 (FI restriction). *Let $\Gamma\lfloor_M$ be a $\Gamma' \sqsubseteq \Gamma$ such that $|\Gamma'| = sup(M)$ and that $\forall 1 \leq i \leq |\Gamma'|$, $\Gamma'_i \neq \omega$ iff $\underline{i} \in FI(M)$.*

Now we state the subject reduction property for β-contraction, using the concept introduced above.

Theorem 1 (SR for β-contraction in λ_{dB}^{sM}). *If $(\lambda.M\ N):\langle \Gamma \vdash_{\lambda_{dB}^{sM}} \tau\rangle$ then $\{\underline{1}/N\}M:\langle \Gamma\lfloor_{\{\underline{1}/N\}M} \vdash_{\lambda_{dB}^{sM}} \tau\rangle$.*

Proof. By case analysis of $(\lambda.M\ N) : \langle \Gamma \vdash_{\lambda_{dB}^{sM}} \tau\rangle$. Note that there are only two possibilities for the last inference step, the rules $\to'_e$ and $\to_e$. We present here the case when $\to'_e$ is the last rule applied. Hence, $\lambda.M : \langle \Gamma \vdash \omega \to \tau\rangle$ and $N : \langle \Delta \vdash \sigma\rangle$ for some context Δ and type σ. If $\Gamma = nil$ then $M : \langle nil \vdash \tau\rangle$. Hence, by a substitution lemma one has that $\{\underline{1}/N\}M : \langle nil \vdash \tau\rangle$. Note that $FI(\{\underline{1}/N\}M) = FI(M) = \emptyset$ thus $(nil \wedge \Delta)\lfloor_{\{\underline{1}/N\}M} = nil$. The proof when $\Gamma \neq nil$ is similar.

Since the type information lost during β-contraction can affect the type as well, we would need a subtyping relation, and an associated inference rule, in order to obtain the SR property for β-reduction.

$$(\omega\text{-}\varphi)\ \frac{M:\langle\Gamma\vdash\tau\rangle}{\varphi_k^i M:\langle\Gamma_{\le k}.\omega^{\underline{i-1}}.\Gamma_{>k}\vdash\tau\rangle},|\Gamma|>k \quad (\omega\text{-}\sigma)\ \frac{N:\langle\Delta\vdash\rho\rangle \qquad M:\langle\Gamma\vdash\tau\rangle}{M\sigma^i N:\langle\Gamma_{<i}.\Gamma_{>i}\vdash\tau\rangle},\Gamma_i=\omega$$

$$(nil\text{-}\varphi)\ \frac{M:\langle\Gamma\vdash\tau\rangle}{\varphi_k^i M:\langle\Gamma\vdash\tau\rangle},|\Gamma|\le k \qquad (nil\text{-}\sigma)\ \frac{N:\langle\Delta\vdash\rho\rangle \qquad M:\langle\Gamma\vdash\tau\rangle}{M\sigma^i N:\langle\Gamma\vdash\tau\rangle},|\Gamma|<i$$

$$(\wedge\text{-}nil\text{-}\sigma)\ \frac{N:\langle nil\vdash\sigma_1\rangle\dots N:\langle nil\vdash\sigma_m\rangle \qquad M:\langle\omega^{\underline{i-1}}.\wedge_{j=1}^m\sigma_j.nil\vdash\tau\rangle}{M\sigma^i N:\langle nil\vdash\tau\rangle}$$

$$(\wedge\text{-}\omega\text{-}\sigma)\ \frac{N:\langle nil\vdash\sigma_1\rangle\dots N:\langle nil\vdash\sigma_m\rangle \qquad M:\langle\Gamma\vdash\tau\rangle}{M\sigma^i N:\langle\Gamma_{<(i-k)}.nil\vdash\tau\rangle},\ \Gamma_i=\wedge_{j=1}^m\sigma_j\ (*)$$

$$(\wedge\text{-}\sigma)\ \frac{N:\langle\Delta^1\vdash\sigma_1\rangle\dots N:\langle\Delta^m\vdash\sigma_m\rangle \qquad M:\langle\Gamma\vdash\tau\rangle}{M\sigma^i N:\langle(\Gamma_{<i}.\Gamma_{>i})\wedge\omega^{\underline{i-1}}.(\Delta^1\wedge\dots\wedge\Delta^m)\vdash\tau\rangle},\ \Gamma_i=\wedge_{j=1}^m\sigma_j\ (**)$$

(*) $\Gamma=\Gamma_{<(i-k)}.\omega^{\underline{k}}.\wedge_{j=1}^m\sigma_j.nil$ and $\Gamma_{(i-k-1)}\neq\omega$ (**) $\Delta^k\neq nil$, for some $1\le k\le m$, or $\Gamma_{>i}\neq nil$

Fig. 2. Typing rules of the system λs^{SM}

3 An intersection type system for λs_e

In order to have an intersection type system for the λs_e-calculus, we introduce a system for λs as a first step. While the type system for λs is based on the system λ_{dB}^{SM}, the system proposed for λs_e is based on the system $\lambda_{dB}^{\wedge}$.

3.1 The system λs^{SM}

Definition 13 (The system λs^{SM}). *The system λs^{SM} is the extension of system λ_{dB}^{SM}, introduced in Definition 10, by the rules presented in Figure 2.*

Observe that, compared with the simple type system for λs and λs_e, which introduces one type inference rule for each operator (cf. [AK01]), there are multiple rules introduced in Figure 2 for the σ and φ operators. This multiplicity reproduces the cases for the updating and substitution lemmas for λ_{dB}^{SM}. For instance, the rule (nil-φ) maintains the same context, since the updating operator will not affect any of the available indices of the corresponding term. Hence, we have a relevance property related to $AI(M)$ instead of $FI(M)$, as stated below.

Lemma 5 (Relevance for λs^{SM}). *If $M:\langle\Gamma\vdash_{\lambda s^{SM}}\tau\rangle$, then $|\Gamma|=sav(M)$ and $\forall 1\le i\le|\Gamma|$, $\Gamma_i\neq\omega$ iff $\underline{i}\in AI(M)$.*

Proof. By induction on the derivation of $M:\langle\Gamma\vdash_{\lambda s^{SM}}\tau\rangle$. We present the case for the application of the rule (nil-φ). Hence, $\varphi_k^i M:\langle\Gamma\vdash\tau\rangle$ where $M:\langle\Gamma\vdash\tau\rangle$ and $|\Gamma|\le k$. By the induction hypothesis (IH) one has that $|\Gamma|=sav(M)$ and $\forall 1\le j\le|\Gamma|$, $\Gamma_j\neq\omega$ iff $\underline{j}\in AI(M)$. Observe that $AI(\varphi_k^i M)=AI(M)_{\le k}\cup(AI(M)_{>k}+(i-1))=AI(M)$ thus $sav(\varphi_k^i M)=sav(M)$.

Despite the fact that the type system is relevant, we have SR for the full s-calculus.

Theorem 2 (SR for s in λs^{SM}). *Let $M : \langle \Gamma \vdash_{\lambda s^{SM}} \tau \rangle$. If $M \rightarrow_s M'$, then $M' : \langle \Gamma \vdash_{\lambda s^{SM}} \tau \rangle$.*

Proof. By the verification of SR for each rewriting rule of the s-calculus.

Observe that the type information associated to the empty application disappears when it becomes an empty substitution, since the rules (nil-σ) and (ω-σ) discard the corresponding contexts. Therefore, we need a restriction notion similar to the one introduced in Definition 12, which is related to the available indices, to have an SR statement for the simulation of β-contraction.

Definition 14 (AI restriction). *Let $\Gamma \upharpoonright_M$ be a $\Gamma' \sqsubseteq \Gamma$ such that $|\Gamma'| = sav(M)$ and that $\forall 1 \leq i \leq |\Gamma'|$, $\Gamma'_i \neq \omega$ iff $\underline{i} \in AI(M)$.*

Theorem 3 (SR for simulation of β-contraction in λs^{SM}). *If $(\lambda.M\ M') : \langle \Gamma \vdash_{\lambda s^{SM}} \tau \rangle$, then $\{\underline{1}/M'\}M : \langle \Gamma\upharpoonright_{\{\underline{1}/M'\}M} \vdash_{\lambda s^{SM}} \tau \rangle$, for any $(\lambda.M\ M') \in \Lambda_{dB}$.*

Proof. The proof consists in the verification of SR with context restriction for $(\lambda.M\ M') : \langle \Gamma \vdash_{\lambda s^{SM}} \tau \rangle$ when the rule (σ-generation) is applied and then of SR for the s-calculus.

3.2 The system $\lambda s_e{}^{\wedge}$

While the λs-calculus has the preservation of strong normalisation property [KR95], PSN for short, the rules allowing the composition of substitution in the λs_e-calculus invalidate this property for the calculus. B. Guillaume presents in [Guillaume2000] a counter example of some simply typed term in λs_e which has an infinite reduction strategy. We present an example below, to give an intuition on how to change the system λs^{SM} to have an intersection type system for λs_e with the subject reduction property.

Example 2. Let $A \equiv (\underline{1}\ \underline{1})$, $M \equiv (\underline{3}\,\sigma^1 A)\,\sigma^1 \lambda.A$, $M' \equiv (\underline{3}\,\sigma^2 \lambda.A)\,\sigma^1 (A\,\sigma^1 \lambda.A)$. We have that $M \rightarrow_{\lambda s_e} M'$, where M is typable in λs^{SM} and M' is not typable. Observe that one cannot obtain M' from M in λs and that M is obtained from the term $M_0 \equiv (\lambda.(\lambda.\underline{3}\ A)\ \lambda.A)$ in both calculi. □

The non typability of the term M_0 above in the system λs^{SM} is due to the inclusion of type information from the context of an argument to an empty application. Note that the typability of both M_0 and $A\,\sigma^1 \lambda.A$ reduces to the typability of $\Omega \equiv (\lambda.A\ \lambda.A)$ which has no type in systems like the Barendregt et al. [BCD83] other then the universal ω type. Hence, we drop the typability requirement on rules $\rightarrow'_e$, (nil-σ) and (ω-σ) , obtaining the system $\lambda s_e{}^{\wedge}$ below.

$$\frac{}{\underline{1}:\langle \tau.nil \vdash \tau\rangle}\ \text{var} \qquad \frac{\underline{n}:\langle \Gamma \vdash \tau\rangle}{\underline{n+1}:\langle \omega.\Gamma \vdash \tau\rangle}\ \text{varn} \qquad \frac{M:\langle u.\Gamma \vdash \tau\rangle}{\lambda.M:\langle \Gamma \vdash u \to \tau\rangle}\ \to_i$$

$$\frac{M_1:\langle \Gamma \vdash \omega \to \tau\rangle}{(M_1\ M_2):\langle \Gamma \vdash \tau\rangle}\ \to_e^{\omega} \qquad \frac{M:\langle nil \vdash \tau\rangle}{\lambda.M:\langle nil \vdash \omega \to \tau\rangle}\ \to_i'$$

$$\frac{M_1:\langle \Gamma \vdash \wedge_{i=1}^n \sigma_i \to \tau\rangle \qquad M_2:\langle \Delta^1 \vdash \sigma_1\rangle\ \ldots\ M_2:\langle \Delta^n \vdash \sigma_n\rangle}{(M_1\ M_2):\langle \Gamma \wedge \Delta^1 \wedge \cdots \wedge \Delta^n \vdash \tau\rangle}\ \to_e$$

$$(nil\text{-}\sigma)\ \frac{M:\langle \Gamma \vdash \tau\rangle}{M\sigma^i N:\langle \Gamma \vdash \tau\rangle},\ |\Gamma| < i \qquad (\omega\text{-}\sigma)\ \frac{M:\langle \Gamma \vdash \tau\rangle}{M\sigma^i N:\langle \Gamma_{<i}.\Gamma_{>i} \vdash \tau\rangle},\ \Gamma_i = \omega$$

$$(\wedge\text{-}nil\text{-}\sigma)\ \frac{N:\langle nil \vdash \sigma_1\rangle \ldots N:\langle nil \vdash \sigma_m\rangle \qquad M:\langle \omega^{\underline{i-1}}.\wedge_{j=1}^m \sigma_j.nil \vdash \tau\rangle}{M\sigma^i N:\langle nil \vdash \tau\rangle}$$

$$(\wedge\text{-}\omega\text{-}\sigma)\ \frac{N:\langle nil \vdash \sigma_1\rangle \ldots N:\langle nil \vdash \sigma_m\rangle \qquad M:\langle \Gamma \vdash \tau\rangle}{M\sigma^i N:\langle \Gamma_{<(i-k)}.nil \vdash \tau\rangle},\ \Gamma_i = \wedge_{j=1}^m \sigma_j\ (*)$$

$$(\wedge\text{-}\sigma)\ \frac{N:\langle \Delta^1 \vdash \sigma_1\rangle \ldots N:\langle \Delta^m \vdash \sigma_m\rangle \qquad M:\langle \Gamma \vdash \tau\rangle}{M\sigma^i N:\langle (\Gamma_{<i}.\Gamma_{>i}) \wedge \omega^{\underline{i-1}}.(\Delta^1 \wedge \cdots \wedge \Delta^m) \vdash \tau\rangle},\ \Gamma_i = \wedge_{j=1}^m \sigma_j\ (**)$$

$$(\omega\text{-}\varphi)\ \frac{M:\langle \Gamma \vdash \tau\rangle}{\varphi_k^i M:\langle \Gamma_{\le k}.\omega^{\underline{i-1}}.\Gamma_{>k} \vdash \tau\rangle},\ |\Gamma| > k \qquad (nil\text{-}\varphi)\ \frac{M:\langle \Gamma \vdash \tau\rangle}{\varphi_k^i M:\langle \Gamma \vdash \tau\rangle},\ |\Gamma| \le k$$

(*) $\Gamma = \Gamma_{<(i-k)}.\omega^{\underline{k}}.\wedge_{j=1}^m \sigma_j.nil$ and $\Gamma_{(i-k-1)} \neq \omega$ (**) $\Delta^k \neq nil$, for some $1 \le k \le m$, or $\Gamma_{>i} \neq nil$

Fig. 3. Typing rules of the system $\lambda s_e{}^{\wedge}$

Definition 15 (The system $\lambda s_e{}^{\wedge}$). *The inference rules for $\lambda s_e{}^{\wedge}$ are given by the rules of the system λ_{dB}^{SM} in Figure 1 and the system λs^{SM} in Figure 2, where the inference rule $\to_e'$, $(nil\text{-}\sigma)$ and $(\omega\text{-}\sigma)$ are replaced by the rules below:*

$$\frac{M:\langle \Gamma \vdash \omega \to \tau\rangle}{(M\ N):\langle \Gamma \vdash \tau\rangle}\ \to_e^{\omega} \qquad (nil\text{-}\sigma)\ \frac{M:\langle \Gamma \vdash \tau\rangle}{M\sigma^i N:\langle \Gamma \vdash \tau\rangle},\ |\Gamma| \le i$$

$$(\omega\text{-}\sigma)\ \frac{M:\langle \Gamma \vdash \tau\rangle}{M\sigma^i N:\langle \Gamma_{<i}.\Gamma_{>i} \vdash \tau\rangle},\ \Gamma_i = \omega$$

The system $\lambda s_e{}^{\wedge}$ is presented in Figure 3.

The system $\lambda s_e{}^{\wedge}$ does not have a defined correspondence relating some syntactic characterisation and relevance. However, the system has a property related to relevance, stated below.

Lemma 6. *If $M:\langle \Gamma \vdash_{\lambda s_e{}^{\wedge}} \tau\rangle$ for $|\Gamma| = m > 0$, then $\Gamma_m \neq \omega$ and $\forall 1 \le i \le m$, $\Gamma_i \neq \omega$ implies $\underline{i} \in AI(M)$.*

Proof. By induction on the derivation of $M:\langle \Gamma \vdash_{\lambda s_e{}^{\wedge}} \tau\rangle$ when $\Gamma \neq nil$.

$$\frac{}{\underline{1}:\langle \tau.nil \vdash \tau\rangle}\ (\text{var}) \qquad \frac{M:\langle u.\Gamma \vdash \tau\rangle}{\lambda.M:\langle \Gamma \vdash u \to \tau\rangle}\ \to_i$$

$$\frac{M_1:\langle \Gamma \vdash \omega \to \tau\rangle}{(M_1\ M_2):\langle \Gamma \vdash \tau\rangle}\ \to_e^{\omega} \qquad \frac{M:\langle nil \vdash \tau\rangle}{\lambda.M:\langle nil \vdash \omega \to \tau\rangle}\ \to_i'$$

$$\frac{M_1:\langle \Gamma \vdash \wedge_{i=1}^{m}\sigma_i \to \tau\rangle \qquad M_2:\langle \Delta^1 \vdash \sigma_1\rangle \ \ldots\ M_2:\langle \Delta^m \vdash \sigma_m\rangle}{(M_1\ M_2):\langle \Gamma \wedge \Delta^1 \wedge \cdots \wedge \Delta^m \vdash \tau\rangle}\ \to_e$$

$$(\text{clos})\ \frac{S:\langle \Gamma \rhd \Gamma'\rangle \qquad M:\langle \Gamma' \vdash \tau\rangle}{M[S]:\langle \Gamma \vdash \tau\rangle}$$

$$(\wedge\text{-cons})\ \frac{M:\langle \Delta^1 \vdash \sigma_1\rangle\ \ldots\ M:\langle \Delta^m \vdash \sigma_m\rangle \qquad S:\langle \Delta \rhd \Delta'\rangle}{M.S:\langle \Delta \wedge \Delta^1 \wedge \cdots \wedge \Delta^m \rhd (\wedge_{i=1}^{m}\sigma_i).\Delta'\rangle}$$

$$(\text{id})\ \frac{\Gamma \neq \Delta.\omega^{\underline{m}}}{id:\langle \Gamma \rhd \Gamma\rangle} \qquad (\text{comp})\ \frac{S:\langle \Gamma \rhd \Gamma''\rangle \qquad S':\langle \Gamma'' \rhd \Gamma'\rangle}{S' \circ S:\langle \Gamma \rhd \Gamma'\rangle}$$

$$(nil\text{-shift})\ \frac{}{\uparrow:\langle nil \rhd nil\rangle} \qquad (nil\text{-cons})\ \frac{S:\langle \Delta \rhd nil\rangle}{M.S:\langle \Delta \rhd nil\rangle}$$

$$(\omega\text{-shift})\ \frac{\Gamma \neq \Delta.\omega^{\underline{n}}}{\uparrow:\langle \omega.\Gamma \rhd \Gamma\rangle} \qquad (\omega\text{-cons})\ \frac{S:\langle \Delta \rhd \Delta'\rangle}{M.S:\langle \Delta \rhd \omega.\Delta'\rangle},\ \Delta' \neq \omega^{\underline{n}}$$

Fig. 4. The inference rules for the system $\lambda\sigma^{\wedge}$

We can prove the subject reduction property for the λs_e-calculus in a standard way, proving some generation lemmas first, where only the $\Gamma_m \neq \omega$ piece of the statement above is needed. Below, we present the subject reduction theorem.

Theorem 4 (SR for $\lambda s_e{}^{\wedge}$). *If $M:\langle \Gamma \vdash_{\lambda s_e{}^{\wedge}} \tau\rangle$ and $M \to_{\lambda s_e} M'$, then $M':\langle \Gamma \vdash_{\lambda s_e{}^{\wedge}} \tau\rangle$.*

Proof. By the verification of SR for each λs_e rewriting rule.

4 An intersection type system for $\lambda\sigma$

Similar to the intersection type system proposed for λs_e, the type system for $\lambda\sigma$ discards any type information from contexts of terms which are related to empty applications.

Definition 16 (The system $\lambda\sigma^{\wedge}$). *The typing rules for the system $\lambda\sigma^{\wedge}$ are presented in Figure 4, where $m > 0$ and $n \geq 0$.*

The next lemma states the property of the system $\lambda\sigma^{\wedge}$ related to relevance.

Lemma 7. *If $M:\langle \Gamma \vdash_{\lambda\sigma^{\wedge}} \tau\rangle$ and $|\Gamma| = m > 0$, then $\Gamma_m \neq \omega$. In particular, if $S:\langle \Gamma \rhd_{\lambda\sigma^{\wedge}} \Gamma'\rangle$ and $|\Gamma| = m > 0$ then $\Gamma_m \neq \omega$ and if $|\Gamma'| = m' > 0$ then $\Gamma'_{m'} \neq \omega$.*

Proof. By induction on the derivation of $M:\langle \Gamma \vdash_{\lambda\sigma^{\wedge}} \tau\rangle$ when $\Gamma \neq nil$, with subinduction on the derivation of $S:\langle \Gamma \rhd_{\lambda\sigma^{\wedge}} \Gamma'\rangle$ when $\Gamma \neq nil$ or $\Gamma' \neq nil$.

Now we establish the SR property for the $\lambda\sigma$-calculus in this system.

Theorem 5 (SR for $\lambda\sigma^{\wedge}$). *If $M : \langle \Gamma \vdash_{\lambda\sigma^{\wedge}} \tau \rangle$ and $M \rightarrow_{\lambda\sigma} M'$ then $M' : \langle \Gamma \vdash_{\lambda\sigma^{\wedge}} \tau \rangle$. In particular, if $S : \langle \Gamma \triangleright_{\lambda\sigma^{\wedge}} \Gamma' \rangle$ and $S \rightarrow_{\lambda\sigma} S'$ then $S' : \langle \Gamma \triangleright_{\lambda\sigma^{\wedge}} \Gamma' \rangle$.*

Proof. By the verification of SR for each $\lambda\sigma$ rewriting rule.

5 Conclusion

In this paper, we proved the subject reduction property for β-contraction in the system λ_{dB}^{SM} [VAK10], using an adaptation for sequential contexts of the restricted environments, introduced in [KN07] to prove SR in a relevant intersection type system. Then, we introduced intersection type systems for two explicit substitution calculi, the $\lambda\sigma$ and the λs_e, and established that our two new systems satisfy the SR property. The simply typed version of these calculi have applications on the HOU problem [DHK2000,AK01] and, to the best of our knowledge, the IT systems presented here are the first polymorphic type systems proposed for them.

We intend to use the systems presented here as the basic system for studying the PT property in IT systems for both calculi. The PT property allows one to include features in a type system which include separate compilation, data abstraction and smartest recompilation [Jim96]. The system $\lambda_{dB}^{\wedge}$, briefly mentioned at the end of Subsection 2.1, is a de Bruijn version of the system in [SM97], were the PT property for β-nfs described in [SM96a] is extended for any normalisable term. Hence, as a first step towards the PT for explicit substitutions, we need to extend the results presented in [VAK10] to normalisable terms in λ_{dB}. Besides that, we believe that the systems λ_{dB}^{SM} and λs^{SM} are able to provide a characterisation for strongly normalising terms in λ_{dB} and λs, respectively. On the other hand, it seems that $\lambda_{dB}^{\wedge}$, $\lambda s_e{}^{\wedge}$ and $\lambda\sigma^{\wedge}$ can provide a characterisation of weak normalisation for λ_{dB}, λs_e and $\lambda\sigma$, respectively.

References

[ACCL91] M. Abadi, L. Cardelli, P.-L. Curien, and J.-J. Lévy. Explicit Substitutions. *J. func. program.*, 1(4):375–416, 1991.

[AK01] M. Ayala-Rincón and F. Kamareddine. Unification via the λs_e-Style of Explicit Substitution. *Logical journal of the IGPL*, 9(4):489–523, 2001.

[Bakel95] S. van Bakel. Intersection Type Assignment Systems. *Theoret. comput. sci.*, 151:385-435, 1995.

[BCD83] H. Barendregt, M. Coppo, and M. Dezani-Ciancaglini. A filter lambda model and the completeness of type assignment. *J. symbolic logic*, 48:931–940, 1983.

[Barendregt84] H. Barendregt, *The Lambda Calculus: Its Syntax and Semantics.* North-Holland, 1984.

[deBruijn72] N.G. de Bruijn. Lambda-Calculus Notation with Nameless Dummies, a Tool for Automatic Formula Manipulation, with Application to the Church-Rosser Theorem. *Indag. Mat.*, 34(5):381–392, 1972.

[deBruijn78] N.G. de Bruijn. A namefree lambda calculus with facilities for internal definition of expressions and segments. T.H.-Report 78-WSK-03, Technische Hogeschool Eindhoven, Nederland, 1978.

[CDC78] M. Coppo and M. Dezani-Ciancaglini. A new type assignment for lambda-terms. *Archiv für mathematische logik*, 19:139–156, 1978.

[CDC80] M. Coppo and M. Dezani-Ciancaglini. An Extension of the Basic Functionality Theory for the λ-Calculus. *Notre dame j. formal logic*, 21(4):685–693, 1980.

[CDV80] M. Coppo, M. Dezani-Ciancaglini and B. Venneri. Principal Type Schemes and λ-calculus Semantics. In J.P. Seldin and J.R. Hindley (eds), *To H.B. Curry: Essays on combinatory logic, lambda calculus and formalism*, pp. 536–560. Academic Press, 1980.

[CF58] H. B. Curry and R. Feys. *Combinatory Logic*, vol. 1. North Holland, 1958.

[DG94] F. Damiani and P. Giannini. A Decidable Intersection Type System based on Relevance. In *Proc. of TACS94*, *LNCS* 789:707725. Springer-Verlag, 1994.

[DHK2000] G. Dowek, T. Hardin, and C. Kirchner. Higher-order Unification via Explicit Substitutions. *Information and Computation*, 157(1/2):183–235, 2000.

[Guillaume2000] B. Guillaume. The λse-calculus does not preserve strong normalisation. *J. of func. program.*, 10(4):321–325, 2000

[Jim96] T. Jim. What are principal typings and what are they good for?. In *Proc. of POPL'95: Symp. on Principles of Programming Languages*, 42–53, ACM, 1996.

[KN07] F. Kamareddine and K. Nour. A completeness result for a realisability semantics for an intersection type system. *Annals pure and appl. logic*, 146:180–198, 2007.

[KR95] F. Kamareddine and A. Ríos. A λ-calculus à la de Bruijn with Explicit Substitutions. In *Proc. of PLILP'95*, *LNCS* 982:45–62. Springer, 1995.

[KR97] F. Kamareddine and A. Ríos. Extending a λ-calculus with Explicit Substitution which Preserves Strong Normalisation into a Confluent Calculus on Open Terms. *J. of Func. Programming*, 7:395–420, 1997.

[KW04] A.J. Kfoury and J.B. Wells. Principality and type inference for intersection types using expansion variables. *Theoret. comput. sci.*, 311(1–3):1–70, 2004.

[LLDDvB04] S. Lengrand, P. Lescanne, D. Dougherty, M. Dezani-Ciancaglini, and S. van Bakel. Intersection types for explicit substitutions. *Inform. and comput.*, 189(1):1742, 2004.

[Milner78] R. Milner. A theory of type polymorphism in programming. *J. comput. and system sci.*, 17(3):348–375, 1978.

[NGdV94] R. P. Nederpelt, J. H. Geuvers, and R. C. de Vrijer. *Selected papers on Automath*. North-Holland, 1994.

[Pottinger80] G. Pottinger. A type assignment for the strongly normalizable λ-terms. In J.P. Seldin and J. R. Hindley (eds), *To H. B. Curry: Essays on combinatory logic, lambda calculus and formalism*, pp. 561–578. Academic Press, 1980.

[RV84] S. Ronchi Della Rocca and B. Venneri. Principal Type Scheme for an Extended Type Theory. *Theoret. comput. sci.*, 28:151–169, 1984.

[Rocca88] S. Ronchi Della Rocca. Principal Type Scheme and Unification for Intersection Type Discipline. *Theoret. comput. sci.*, 59:181–209, 1988.

[SM96a] E. Sayag and M. Mauny. Characterization of principal type of normal forms in intersection type system. In *Proc. of FSTTCS'96*, LNCS, 1180:335–346. Springer, 1996.

[SM96b] E. Sayag and M. Mauny. A new presentation of the intersection type discipline through principal typings of normal forms. *Tech. rep.* RR-2998, INRIA, 1996.

[SM97] E. Sayag and M. Mauny. Structural properties of intersection types. In *Proc. of LIRA'97*, pp. 167-175. Novi Sad, Yugoslavia, 1997.

[VAK09] D. Ventura and M. Ayala-Rincón and F. Kamareddine. Intersection Type System with de Bruijn Indices. *The many sides of logic.* Studies in logic 21:557–576, College publications. London, 2009.

[VAK10] D. Ventura and M. Ayala-Rincón and F. Kamareddine. Principal Typings in a Restricted Intersection Type System for Beta Normal Forms with de Bruijn Indices. In *Proc. of WRS'09*. EPTCS 15:69–82, 2010.

[Ventura10] D. Ventura. Cálculos de Substituições Explícitas à la de Bruijn com Sistemas de Tipos com Interseção. *PhD Thesis* Departamento de Matemática, Universidade de Brasília, 2010. Online version (in Portuguese) from 5th March at `http://ventura.mat.unb.br`