Challenges and solutions to realisability semantics for intersection types with expansion variables

Fairouz Kamareddine, Karim Nour, Vincent Rahli, J. B. Wells

Fundamenta Informaticae XXI (2001) 1001–1032
IOS Press

On Realisability Semantics for Intersection Types with Expansion Variables

Fairouz Kamareddine

ULTRA Group (Useful Logics, Types, Rewriting, and their Automation), Heriot-Watt University, School of Mathematical and Computer Sciences, Edinburgh EH14 4AS, UK.

Email: http://www.macs.hw.ac.uk/ultra/

Karim Nour

Université de Savoie, Campus Scientifique, 73378 Le Bourget du Lac, France.

Email: nour@univ-savoie.fr

Vincent Rahli[*]

and J. B. Wells[†]

Abstract. Expansion is a crucial operation for calculating principal typings in intersection type systems. Because the early definitions of expansion were complicated, *E-variables* were introduced in order to make the calculations easier to mechanise and reason about. Recently, E-variables have been further simplified and generalised to also allow calculating other type operators than just intersection. There has been much work on semantics for type systems with intersection types, but none whatsoever before our work, on type systems with E-variables. In this paper we expose the challenges of building a semantics for E-variables and we provide a novel solution. Because it is unclear how to devise a space of meanings for E-variables, we develop instead a space of meanings for types that is hierarchical. First, we index each type with a natural number and show that although this intuitively captures the use of E-variables, it is difficult to index the universal type ω with this hierarchy and it is not possible to obtain completeness of the semantics if more than one E-variable is used. We then move to a more complex semantics where each type is associated with a list of natural numbers and establish that both ω and an arbitrary number of E-variables can be represented without losing any of the desirable properties of a realisability semantics.

Keywords: Realisability semantics, expansion variables, intersection types, completeness

Address for correspondence: ULTRA Group (Useful Logics, Types, Rewriting, and their Automation), Heriot-Watt University, School of Mathematical and Computer Sciences, Mountbatten building, Edinburgh EH14 4AS, UK. Email: http://www.macs.hw.ac.uk/ultra/

[*] Same address as Kamareddine.

[†] Same address as Kamareddine.

1. Introduction

Intersection types and the expansion mechanism. Intersection types were developed in the late 1970s to type λ-terms that are untypable with simple types; they do this by providing a kind of finitary type polymorphism where the usages (types) of terms are listed rather than obtained by quantification. They have been useful in reasoning about the semantics of the λ-calculus, and have been investigated for use in static program analysis. *Expansion* was introduced at the end of the 1970s as a crucial procedure for calculating *principal typings* for λ-terms in type systems with intersection types, allowing support for compositional type inference. Coppo, Dezani, and Venneri [7] introduced the operation of *expansion* on *typings* (pairs of a type environment and a result type) for calculating the possible typings of a term when using intersection types. As a simple example, there exists an intersection type system S where the λ-term $M = (\lambda x.x(\lambda y.yz))$ can be assigned the typing $\Phi_1 = \langle\{z \mapsto a\}, (((a\to b)\to b)\to c)\to c\rangle$, which happens to be its principal typing in S. The term M can also be assigned the typing $\Phi_2 = \langle s\{z \mapsto a_1 \sqcap a_2\}, ((((a_1\to b_1)\to b_1) \sqcap ((a_2\to b_2)\to b_2))\to c)\to c\rangle$, and an expansion operation can yield Φ_2 from Φ_1.

Expansion variables. Because the early definitions of expansion were complicated, *E-variables* were introduced in order to make the calculations easier to mechanize and reason about. For example, in System E [5], the typing Φ_1 presented above is replaced by $\Phi_3 = \langle\{z \mapsto ea\}, ((e((a\to b)\to b))\to c)\to c\rangle$, which differs from Φ_1 by the insertion of the E-variable e at two places (in both components of the Φ_3), and Φ_2 can be obtained from Φ_3 by substituting for e the *expansion term* $E = (a := a_1, b := b_1) \sqcap (a := a_2, b := b_2)$. Carlier and Wells [6] have surveyed the history of expansion and also E-variables.

Designing a space of meanings for expansion variables. In many kinds of semantics, a type T is interpreted by a second order function $[T]_\nu$ that takes two parameters, the type T and also a valuation ν that assigns to type variables the same kind of meanings that are assigned to types. To extend this idea to types with E-variables, we need to devise some space of possible meanings for E-variables. Given that a type eT can be turned by expansion into a new type $S_1(T) \sqcap S_2(T)$, where S_1 and S_2 are arbitrary substitutions (which can themselves introduce expansions), and that this can introduce an unbound number of new variables (both E-variables and regular type variables), the situation is complicated. Because it is unclear how to devise a space of meanings for expansions and E-variables, we instead restrict ourselves to E-variables and develop a space of meanings for types that is hierarchical in the sense that we can split it w.r.t. a certain concept of degree. Although this idea is not perfect, it seems to go quite far in giving an intuition for E-variables, namely that each E-variable occurring in a typing associated with a λ-term, acts as a capsule that isolates parts of the λ-term. As future work, we wish to come up with a higher order function that interprets types involving expansion terms by sets of λ-terms. We believe this function would help regarding the substitution mechanism introduced by expansion in terms of λ-expressions.

Our semantic approach. The semantic approach we use in the current document is a realisability semantics in the sense that it is derived from Kreisel's modified realisability and its variants, where "a formula "x realizes A" can be defined in a completely straightforward way: the type of the variable x is determined by the logical form of A" [26], x being the code of a function. Our semantics is strongly related to the semantic argument used in reducibility methods as used and developed by Tait [27] and many others after him [24, 23, 13, 12, 14, 15]. Atomic types (e.g., type variables) are interpreted as *saturated* sets of λ-terms, meaning that they are closed under β-expansion (the inverse of β-reduction). Arrow types are interpreted by function spaces (see the semantics provided by Scott in the open problems published in the proceedings of the Lecture Notes in Computer Science symposium held in 1975 [4]) and intersection types are interpreted by set intersections. Such a realisability semantics allows one to prove *soundness* w.r.t. a type system S, i.e., the meaning of a type T contains all closed λ-terms that can be assigned T in S. This has been shown useful for characterising the behaviour of typed λ-terms [24]. One also wants to show the converse of soundness which is called *completeness*, i.e., every closed λ-term in the meaning of T can be assigned T in S.

Completeness results. Hindley [17, 18, 19] was one of the first to investigate such completeness results for a simple type system and he showed that all the types of that system have the completeness property. He then generalised his completeness proof to an intersection type system [16]. Using his completeness theorem based on saturated sets of λ-terms w.r.t. $\beta\eta$-equivalence, Hindley showed that simple types were "realised"[1] by all and only the λ-terms which are typable by these types. Note that Hindley's completeness theorems were established with the sets of λ-terms saturated by $\beta\eta$-equivalence. In the present document, our completeness result depends only on the weaker requirement of β-equivalence, and we have managed to make simpler proofs that avoid needing η-reduction, confluence, or SN (although we do establish both confluence and SN for both β and $\beta\eta$).

Similar approaches to type interpretation. Recent works on realisability related to ours include that by Labib-Sami [25], Farkh and Nour [11], and Coquand [9], although none of this work deals with intersection types or E-variables. Similar work on realisability dealing with intersection types includes that by Kamareddine and Nour [21], which gives a sound and complete realisability semantics w.r.t. an intersection type system. This system does not deal with E-variables and is therefore different from the three hierarchical systems presented in this document. The main difference is the hierarchies which did not exist in Kamareddine and Nour's document [21].

Towards a semantics of expansion. Initially, we aimed to give a realisability semantics for a system of expansions proposed by Carlier and Wells [6]. In order to simplify our study, we considered the system with expansion variables but without the expansion rewriting rules (without the expansion mechanism). In essence, this meant that the type syntax was: $T \in \mathsf{Ty} ::= a \mid \omega \mid T_1 \to T_2 \mid T_1 \sqcap T_2 \mid eT$ where a is a type variable ranging over a countably infinite type variable set TyVar and e is an expansion variable ranging over a countably infinite expansion variable set ExpVar, and that the typing rules were as follows:

$$\frac{}{x : \langle \{x \mapsto T\} \vdash T\rangle}\ (\mathsf{var}) \qquad \frac{}{M : \langle \varnothing \vdash \omega\rangle}\ (\omega)$$

$$\frac{M : \langle \Gamma \uplus \{x \mapsto T_1\} \vdash T_2\rangle}{\lambda x.M : \langle \Gamma \vdash T_1 \to T_2\rangle}\ (\mathsf{abs}) \qquad \frac{M_1 : \langle \Gamma_1 \vdash T_1 \to T_2\rangle \quad M_2 : \langle \Gamma_2 \vdash T_1\rangle}{M_1 M_2 : \langle \Gamma_1 \sqcap \Gamma_2 \vdash T_2\rangle}\ (\mathsf{app})$$

$$\frac{M : \langle \Gamma_1 \vdash T_1\rangle \quad M : \langle \Gamma_2 \vdash T_2\rangle}{M : \langle \Gamma_1 \sqcap \Gamma_2 \vdash T_1 \sqcap T_2\rangle}\ (\sqcap) \qquad \frac{M : \langle \Gamma \vdash T\rangle}{M : \langle e\Gamma \vdash eT\rangle}\ (\mathsf{e\text{-}app})$$

To provide a realisability semantics for this system, we needed to define the interpretation of a type to be a set of terms having this type. For our semantics to be informative on expansion variables, we needed to distinguish between the interpretation of T and eT. However, in the typing rule (e-app)

[1] We say that a λ-term M "realises" a type T if M is in T's interpretation. Hindley's semantics is not a realisability semantics but it bears some resemblance with modified realisability. One of Hindley's semantics is called "the simple semantics" and is based on the concept of model of the untyped λ-calculus [20]. Our type interpretation is also similar to Hindley's[16].

presented above, the term M is unchanged and this poses difficulties. For this reason, we modified slightly the above type system by indexing the terms of the λ-calculus giving us the following syntax of terms: $M ::= x^i \mid (MN) \mid (\lambda x^i.M)$ (where M and N need to satisfy a certain condition before (MN) is allowed to be a term) and by slightly changing our type rules and in particular rule (e-app):

$$\frac{M : \langle \Gamma \vdash U \rangle}{M^+ : \langle e\Gamma \vdash eU \rangle}\ (\mathsf{e\text{-}app})$$

In this new (e-app) rule, M^+ is M where all the indices are increased by 1. Obviously these indices needed a revision regarding β-reduction and the typing rules in order to preserve the desirable properties of the type system and the realisability semantics. For this, we defined the good terms and the good types and showed that these notions go hand in hand (e.g., good types can only be assigned to good terms).

We developed a realisability semantics where each use of an E-variable in a type corresponds to an index at which evaluation occurs in the λ-terms that are assigned the type. This was an elegant solution that captured the intuition behind E-variables. However, in order for this new type system to behave well, it was necessary to consider λI-terms only (removing a subterm from M also removes important information about M as in the reduction $(\lambda x.y)M \rightarrow_\beta y$ where M is thrown away). It was also necessary to drop the universal type ω completely. This led us to the introduction of the $\lambda I^{\mathbb{N}}$-calculus and to our first type system $\vdash_1$ for which we developed a sound realisability semantics for E-variables.

However, although the first type system $\vdash_1$ is crucial to understand the intuition behind out indexed calculus, the realisability semantics we proposed was not complete w.r.t. $\vdash_1$ (subject reduction does not hold either). For this reason, we modified our system $\vdash_1$ by considering a smaller set of types (where intersections and expansions cannot occur directly to the right of an arrow), and by adding subtyping rules. This new type system $\vdash_2$ has subject reduction. Our semantics turned out to be sound w.r.t. $\vdash_2$. As for completeness, we needed to limit the list of expansion variables to a single element list. This completeness issue for $\vdash_2$ comes from the fact that the natural numbers as indexes do not allow one to differentiate between the types e_1T and e_2T if $e_1 \neq e_2$. Again, we were forced to revise our type system. We decided to restrict our λ-terms by indexing them by lists of natural numbers (where each natural number represents a difference expansion variable). We updated the type system $\vdash_2$ in consequence to obtain the type system $\vdash_3$ based among other things on the following new (e-app) rule:

$$\frac{M : \langle \Gamma \vdash U \rangle}{M^{+i} : \langle e\Gamma \vdash eU \rangle}\ (\mathsf{e\text{-}app})$$

where i is the natural number associated with the expansion variable e and where M^{+i} is M where all the lists of natural numbers are augmented with i. This new rule (e-app) allows us to distinguish the interpretations of the types e_1T and e_2T when $e_1 \neq e_2$. Furthermore, our λ-terms are constructed in such a way that K-reductions do not limit the information on the reduced terms (as in the $\lambda I^{\mathbb{N}}$-calculus, β-reduction is not always allowed, and in addition we impose further restriction on applications and abstractions). In order to obtain completeness in presence of the ω-rule, we also consider ω indexed by lists. This means that the new calculus becomes rather heavy but this seems unavoidable. It is needed to obtain a complete realisability semantics where an arbitrary (possibly infinite) number of expansion variables is allowed and where ω is present. The use of lists complicates matters and hence, needs to be understood in the context of the first semantics where indices are natural numbers rather than lists of natural numbers. In addition to the above, we consider three saturation notions (in line with the literature) illustrating that these notions behave well in our complete realisability semantics.

Road map. Sec. 2.1 gives the syntax of the indexed calculi considered in this document: the $\lambda I^{\mathbb{N}}$-calculus, which is the λI-calculus with each variable annotated by a natural number called a *degree* or *index*, and the $\lambda^{\mathcal{L}_{\mathbb{N}}}$-calculus which is the full λ-calculus (where K-redexes are allowed) indexed with finite sequences of natural numbers. We show the confluence of β, $\beta\eta$ and weak head h-reduction on our indexed λ-calculi. Sec. 2.2 introduces the syntax and terminology for types used in both indexed calculi. Sec. 2.3 introduces our three intersection type systems with E-variables $\vdash_i$ for $i \in \{1, 2, 3\}$, where in the first one, the syntax of types is not restricted (and hence subject reduction fails) but in the other two it is restricted but then the systems are extended with a subtyping relation. In Sec. 2.4.1 and Sec. 2.4.2 we study the properties of our three type systems including subject reduction and expansion with respect to our various reduction relations ($\beta, \beta\eta, h$). Sec. 3.1 introduces our realisability semantics and show its soundness w.r.t. each of the three considered type systems (and for each reduction relation). Sec. 3.2 discusses the challenges of showing completeness of the realisability semantics designed for the first two systems. We show that completeness does not hold for the first system and that it also does not hold for the second system if more than one expansion variable is used, but does hold for a restriction of this system to one single E-variable. This is already an important step in the study of the semantics of intersection type systems with expansion variables since a single expansion variable can be used many times and can occur nested. Sec. 3.3 establishes the completeness of a given realisability semantics w.r.t. $\vdash_3$ by introducing a special interpretation. We conclude in Sec. 4 and proofs are presented in the expanded version of this article [22].

2. The $\lambda I^{\mathbb{N}}$ and $\lambda^{\mathcal{L}_{\mathbb{N}}}$ calculi and associated type systems

2.1. The syntax of the indexed λ-calculi

Definition 2.1. (Indices)
We introduce two kinds of indices: natural numbers for our first semantics and sequences of natural numbers for our second semantics. Let $\mathcal{L}_{\mathbb{N}} = \mathsf{tuple}(\mathbb{N})$. We let I, J, range over indices. The metavariables I and J will range over $\mathbb{N}$ when considering the $\lambda I^{\mathbb{N}}$-calculus and over $\mathcal{L}_{\mathbb{N}}$ when considering the $\lambda^{\mathcal{L}_{\mathbb{N}}}$-calculus (both these calculus are defined below). Let L, K, R range over $\mathcal{L}_{\mathbb{N}}$. We sometimes write $\langle n_1, \ldots, n_m \rangle$ as $(n_1, \ldots, n_m)$ or as $(n_i)_{1 \le i \le m}$ or as $(n_i)_m$. We denote $\oslash$ the empty sequence of natural numbers ($\oslash$ stands for $\langle\rangle$). Let $::$ add an element to a sequence: $j :: (n_1, \ldots, n_m) = (j, n_1, \ldots, n_m)$. We sometimes write $L_1 @ L_2$ as $L_1 :: L_2$. We define the relation $\preceq$ and $\succeq$ on $\mathcal{L}_{\mathbb{N}}$ as follows: $L_1 \preceq L_2$ (or $L_2 \succeq L_1$) iff there exists $L_3 \in \mathcal{L}_{\mathbb{N}}$ such that $L_2 = L_1 :: L_3$.

Lemma 2.1. $\preceq$ is a partial order on $\mathcal{L}_{\mathbb{N}}$.

Let x, y, z range over Var, a countable infinite set of term variables (or just variables).

We define below two indexed calculi: the $\lambda I^{\mathbb{N}}$-calculus (whose set of terms is $\mathcal{M}_1$ as well as $\mathcal{M}_2$ for notational reasons) and the $\lambda^{\mathcal{L}_{\mathbb{N}}}$-calculus (whose set of terms is $\mathcal{M}_3$). As obvious, indices in $\lambda I^{\mathbb{N}}$ are simple but only allow the I-part of the calculus.

We let M, N, P, Q, R range over any of $\mathcal{M}_1$, $\mathcal{M}_2$, and $\mathcal{M}_3$ (we make explicit when a term is taken from either one of these sets). We use $=$ for syntactic equality. We assume the usual definition of subterms and the usual convention for parentheses and their omission (see Barendregt [2] and Krivine [24]). We also consider in this part an extension of the function fv that gathers the indexed λ-term variables occurring free in terms (redefined below).

The joinability $M \diamond N$ of terms M and N ensures that in any term in which M and N occur, each variable has a unique index (note that it is more accurate to include this as part of the simultaneous inductions in Def. 2.3 and 2.5 defining $\mathcal{M}_1$, $\mathcal{M}_2$, and $\mathcal{M}_3$, but for clarity, we define it separately here).

Definition 2.2. (Joinability $\diamond$)
Let $i \in \{1, 2, 3\}$.

- Let M, N be terms of $\lambda I^{\mathbb{N}}$ (resp. $\lambda^{\mathcal{L}_{\mathbb{N}}}$) and let $\mathsf{fv}(M)$ and $\mathsf{fv}(N)$ be the corresponding free variables. We say that M and N are joinable and write $M \diamond N$ iff for all $x \in \mathsf{Var}$, if $x^{L_1} \in \mathsf{fv}(M)$ and $x^{L_2} \in \mathsf{fv}(N)$ (where $L_1, L_2 \in \mathbb{N}$ (resp. $\in \mathcal{L}_{\mathbb{N}}$)) then $L_1 = L_2$.
- If $\overline{M} \subseteq \mathcal{M}_i$ such that $\forall M, N \in \overline{M}.\ M \diamond N$, we write $\diamond \overline{M}$.
- If $\overline{M} \subseteq \mathcal{M}_i$ and $M \in \mathcal{M}_i$ such that $\forall N \in \overline{M}.\ M \diamond N$, we write $M \diamond \overline{M}$.

Now we give the syntax of $\lambda I^{\mathbb{N}}$, an indexed version of the λI-calculus where indices (which range over $\mathbb{N}$) help categorise the *good terms* where the degree of a function is never larger than that of its argument. This amounts to having the full λI-calculus at each index and creating new λI-terms through a mixing recipe. Note that one could also define $\lambda I^{\mathbb{N}}$ by dividing Var into an countably infinite number of sets and by defining a bijective function that associates a unique index with each of these sets. We did not choose to do so because we believe explicitly writing down indexes to be clearer.

Definition 2.3. (The set of terms $\mathcal{M}_1$ (also called $\mathcal{M}_2$))
The set of terms $\mathcal{M}_1$, $\mathcal{M}_2$ (where $\mathcal{M}_1 = \mathcal{M}_2$), the set of free variables $\mathsf{fv}(M)$ of $M \in \mathcal{M}_2$ and the degree $\mathsf{deg}(M)$ of a term M, are defined by simultaneous induction:

- If $x \in \mathsf{Var}$ and $n \in \mathbb{N}$ then $x^n \in \mathcal{M}_2$, $\mathsf{fv}(x^n) = \{x^n\}$, and $\mathsf{deg}(x^n) = n$.
- If $M, N \in \mathcal{M}_2$ such that $M \diamond N$ (see Def. 2.2) then $MN \in \mathcal{M}_2$, $\mathsf{fv}(MN) = \mathsf{fv}(M) \cup \mathsf{fv}(N)$ and $\mathsf{deg}(MN) = \mathsf{min}(\mathsf{deg}(M), \mathsf{deg}(N))$ (where min returns the smallest of its arguments).
- If $M \in \mathcal{M}_2$ and $x^n \in \mathsf{fv}(M)$ then $\lambda x^n.M \in \mathcal{M}_2$, $\mathsf{fv}(\lambda x^n.M) = \mathsf{fv}(M) \setminus \{x^n\}$, and $\mathsf{deg}(\lambda x^n.M_1) = \mathsf{deg}(M_1)$.

Let $ix \in \mathsf{IVar}_2 ::= x^n$ and $\mathsf{IVar}_1 = \mathsf{IVar}_2$. For each $n \in \mathbb{N}$, let $\mathcal{M}_2^n = \{M \in \mathcal{M}_2 \mid \mathsf{deg}(M) = n\}$. Note that a subterm of $M \in \mathcal{M}_2$ is also in $\mathcal{M}_2$. Closed terms are defined as usual: let $\mathsf{closed}(M)$ be true iff M is closed, i.e., iff $\mathsf{fv}(M) = \varnothing$.

Here is now the syntax of good terms in the $\lambda I^{\mathbb{N}}$-calculus.

Definition 2.4. (The set of good terms $\mathbb{M} \subset \mathcal{M}_2$)

1. The set of good terms $\mathbb{M} \subset \mathcal{M}_2$ is defined by:
   - If $x \in \mathsf{Var}$ and $n \in \mathbb{N}$ then $x^n \in \mathbb{M}$.
   - If $M, N \in \mathbb{M}$, $M \diamond N$, and $\mathsf{deg}(M) \leq \mathsf{deg}(N)$ then $MN \in \mathbb{M}$.
   - If $M \in \mathbb{M}$ and $x^n \in \mathsf{fv}(M)$ then $\lambda x^n.M \in \mathbb{M}$.

   Note that a subterm of $M \in \mathbb{M}$ is also in $\mathbb{M}$.

2. For each $n \in \mathbb{N}$, we let $\mathbb{M}^n = \mathbb{M} \cap \mathcal{M}_2^n$

Lemma 2.2. 1. ($M \in \mathbb{M}$ and $x^n \in \mathsf{fv}(M)$) iff $\lambda x^n.M \in \mathbb{M}$.

2. ($M_1, M_2 \in \mathbb{M}$, $M_1 \diamond M_2$ and $\mathsf{deg}(M_1) \leq \mathsf{deg}(M_2)$) iff $M_1 M_2 \in \mathbb{M}$.

Now, we give the syntax of $\lambda^{\mathcal{L}_\mathbb{N}}$. Note that in $\mathcal{M}_3$, an application MN is only allowed when $\mathsf{deg}(M) \preceq \mathsf{deg}(N)$. This restriction did not exist in $\lambda I^\mathbb{N}$ (in $\mathcal{M}_2$'s definition). Furthermore, we only allow abstractions of the form $\lambda x^L.M$ in $\lambda^{\mathcal{L}_\mathbb{N}}$ when $L \succeq \mathsf{deg}(M)$ (a similar restriction holds in $\lambda I^\mathbb{N}$ since it is a variant of the λI-calculus). The elegance of $\lambda I^\mathbb{N}$ is the ability to give the syntax of good terms, which is not obvious in $\lambda^{\mathcal{L}_\mathbb{N}}$.

Definition 2.5. (The set of terms $\mathcal{M}_3$)
The set of terms $\mathcal{M}_3$, the set of free variables $\mathsf{fv}(M)$ and degree $\mathsf{deg}(M)$ of $M \in \mathcal{M}_3$ are defined by simultaneous induction:

- If $x \in \mathsf{Var}$ and $L \in \mathcal{L}_\mathbb{N}$ then $x^L \in \mathcal{M}_3$, $\mathsf{fv}(x^L) = \{x^L\}$, and $\mathsf{deg}(x^L) = L$.
- If $M, N \in \mathcal{M}_3$, $\mathsf{deg}(M) \preceq \mathsf{deg}(N)$, and $M \diamond N$ (see Def. 2.2) then $MN \in \mathcal{M}_3$, $\mathsf{fv}(MN) = \mathsf{fv}(M) \cup \mathsf{fv}(N)$ and $\mathsf{deg}(MN) = \mathsf{deg}(M)$.
- If $x \in \mathsf{Var}$, $M \in \mathcal{M}_3$, and $L \succeq \mathsf{deg}(M)$ then $\lambda x^L.M \in \mathcal{M}_3$, $\mathsf{fv}(\lambda x^L.M) = \mathsf{fv}(M) \setminus \{x^L\}$ and $\mathsf{deg}(\lambda x^L.M) = \mathsf{deg}(M)$.

Let $ix \in \mathsf{IVar}_3 ::= x^L$. Note that each subterm of $M \in \mathcal{M}_3$ is also in $\mathcal{M}_3$. Closed terms are defined as usual: let $\mathsf{closed}(M)$ be true iff M is closed, i.e., iff $\mathsf{fv}(M) = \varnothing$.

In our systems, expansions change the degree of a term. Therefore we define functions to increase and decrease indexes in terms (see Def. 2.6 and Def. 2.7). Note that both the increasing and the decreasing functions are well behaved operations with respect to all that matters (free variables, reduction, joinability, substitution, etc.).

Definition 2.6. 1. For each $n \in \mathbb{N}$, let $\mathcal{M}_2^{\geq n} = \{M \in \mathcal{M}_2 \mid \mathsf{deg}(M) \geq n\}$ and $\mathcal{M}_2^{>n} = \mathcal{M}_2^{\geq n+1}$.

2. We define $^+$ ($\in \mathcal{M}_2 \to \mathcal{M}_2$) and $^-$ ($\in \mathcal{M}_2^{>0} \to \mathcal{M}_2$) as follows:

$$\begin{array}{lll} (x^n)^+ = x^{n+1} & (M_1M_2)^+ = M_1^+M_2^+ & (\lambda x^n.M)^+ = \lambda x^{n+1}.M^+ \\ (x^n)^- = x^{n-1} & (M_1M_2)^- = M_1^-M_2^- & (\lambda x^n.M)^- = \lambda x^{n-1}.M^- \end{array}$$

3. Let $\overline{M} \subseteq \mathcal{M}_2$. If $\forall M \in \overline{M}.\, \mathsf{deg}(M) > 0$, we write $\mathsf{deg}(\overline{M}) > 0$. Also:

$$(\overline{M})^+ = \{M^+ \mid M \in \overline{M}\} \qquad \text{If } \mathsf{deg}(\overline{M}) > 0,\ (\overline{M})^- = \{M^- \mid M \in \overline{M}\}$$

4. We define M^{-n} by induction on $\mathsf{deg}(M) \geq n > 0$. If $n = 0$ then $M^{-n} = M$ and if $n \geq 0$ then $M^{-(n+1)} = (M^{-n})^-$.

Definition 2.7. Let $i \in \mathbb{N}$ and $M \in \mathcal{M}_3$.

1. For each $L \in \mathcal{L}_{\mathbb{N}}$, let:

$$\mathcal{M}_3^L = \{M \in \mathcal{M}_3 \mid \mathsf{deg}(M) = L\} \qquad \mathcal{M}_3^{\succeq L} = \{M \in \mathcal{M}_3 \mid \mathsf{deg}(M) \succeq L\}$$

2. We define M^{+i} as follows:

$$(x^L)^{+i} = x^{i::L} \qquad (M_1 M_2)^{+i} = M_1^{+i} M_2^{+i} \qquad (\lambda x^L.M)^{+i} = \lambda x^{i::L}.M^{+i}$$

3. If $\mathsf{deg}(M) = i :: L$, we define M^{-i} as follows:

$$(x^{i::L})^{-i} = x^L \qquad (M_1 M_2)^{-i} = M_1^{-i} M_2^{-i} \qquad (\lambda x^{i::L'}.M)^{-i} = \lambda x^{L'}.M^{-i}$$

4. Let $\overline{M} \subseteq \mathcal{M}_3$. Let $(\overline{M})^{+i} = \{M^{+i} \mid M \in \overline{M}\}$.
   Note that $(\overline{M}_1 \cap \overline{M}_2)^{+i} = (\overline{M}_1)^{+i} \cap (\overline{M}_2)^{+i}$.

Definition 2.8. (Substitution, alpha conversion, compatibility, reduction)

- Let $M, N_1, \ldots, N_n$ be terms of $\lambda I^{\mathbb{N}}$ (resp. $\lambda^{\mathcal{L}_{\mathbb{N}}}$) and $I_1, \ldots, I_n \in \mathbb{N}$ (resp. $\mathcal{L}_{\mathbb{N}}$). The simultaneous substitution $M[x_1^{I_1} := N_1, \ldots, x_n^{I_n} := N_n]$ of N_i for all free occurrences of $x_i^{I_i}$ in M, where $i \in \{1, \ldots, n\}$, is defined as a partial substitution satisfying these conditions:
  - $\diamond\overline{M}$ where $\overline{M} = \{M\} \cup \{N_i \mid i \in \{1, \ldots, n\}\}$.
  - $\forall i \in \{1, \ldots, n\}.\ \mathsf{deg}(N_i) = I_i$[2].

  We sometimes write $M[x_1^{I_1} := N_1, \ldots, x_n^{I_n} := N_n]$ as $M[(x_i^{I_i} := N_i)_{1 \le i \le n}]$ (or simply $M[(x_i^{I_i} := N_i)_n]$).

- In $\lambda I^{\mathbb{N}}$ (resp. $\lambda^{\mathcal{L}_{\mathbb{N}}}$), we take terms modulo *α-conversion* given by: $\lambda x^I.M = \lambda y^I.(M[x^I := y^I])$ where $\forall I'.\ y^{I'} \notin \mathsf{fv}(M)$ (where $I, I' \in \mathbb{N}$ (resp. $\mathcal{L}_{\mathbb{N}}$)).

- Let $i \in \{1,2,3\}$. We say that a relation on $\mathcal{M}_i$ is *compatible* iff for all $M, N, P \in \mathcal{M}_i$:
  - (iabs): If $M\ rel\ N$ and $\lambda x^I.M, \lambda x^I.N \in \mathcal{M}_i$ then $(\lambda x^I.M)\ rel\ (\lambda x^I.N)$.
  - (iapp_1): If $M\ rel\ N$ and $MP, NP \in \mathcal{M}_i$ then $MP\ rel\ NP$.
  - (iapp_2): If $M\ rel\ N$, and $PM, PN \in \mathcal{M}_i$ then $PM\ rel\ PN$.

- Let $i \in \{1,2,3\}$. The reduction relation $\rightarrow_\beta$ on $\mathcal{M}_i$ is defined as the least compatible relation closed under the rule: $(\lambda x^I.M)N \rightarrow_\beta M[x^I := N]$ if $\mathsf{deg}(N) = I$.

- Let $i \in \{1,2,3\}$. The reduction relation $\rightarrow_\eta$ on $\mathcal{M}_i$ is defined as the least compatible relation closed under the rule: $\lambda x^I.Mx^I \rightarrow_\eta M$ if $x^I \notin \mathsf{fv}(M)$.

- Let $i \in \{1,2,3\}$. The weak head reduction $\rightarrow_h$ on $\mathcal{M}_i$ is defined as the least relation closed by rule (iapp_2) presented above and also by the following rule: $(\lambda x^I.M)N \rightarrow_h M[x^I := N]$ if $\mathsf{deg}(N) = I$.

[2]We can prove the following lemma: if $\overline{M} = \{M\} \cup \{N_j \mid j \in \{1, \ldots, n\}\}$ then we have ($\diamond\overline{M}$ and $\forall j \in \{1, \ldots, n\}.\ \mathsf{deg}(N_j) = I_j$) iff $M[x_1^{I_1} := N_1, \ldots, x_n^{I_n} := N_n] \in \mathcal{M}_i$ where $i \in \{1,2,3\}$.

- Let $\rightarrowtriangle_{\beta\eta} = \rightarrowtriangle_\beta \cup \rightarrowtriangle_\eta$.
- For a reduction relation $\rightarrowtriangle_r$, we denote by $\rightarrowtriangle_r^*$ the reflexive (w.r.t. $\mathcal{M}_i$) and transitive closure of $\rightarrowtriangle_r$. We denote by $\simeq_r$ the equivalence relation induced by $\rightarrowtriangle_r^*$ (symmetric closure).

The next theorem states that reductions do not introduce new free variables and preserve the degree of a term.

Theorem 2.1. Let $i \in \{1, 2, 3\}$, $M \in \mathcal{M}_i$, and $r \in \{\beta, \beta\eta, h\}$.

1. If $M \rightarrowtriangle_\eta^* N$ then $\mathsf{fv}(N) = \mathsf{fv}(M)$ and $\mathsf{deg}(M) = \mathsf{deg}(N)$.
2. If $i = 3$ and $M \rightarrowtriangle_r^* N$ then $\mathsf{fv}(N) \subseteq \mathsf{fv}(M)$ and $\mathsf{deg}(M) = \mathsf{deg}(N)$.
3. If $i \neq 3$ and $M \rightarrowtriangle_\beta^* N$ then $\mathsf{fv}(M) = \mathsf{fv}(N)$, $\mathsf{deg}(M) = \mathsf{deg}(N)$, and $M \in \mathbb{M}$ iff $N \in \mathbb{M}$.

Proof:
1. By induction on $M \rightarrowtriangle_\eta^* N$. 2. Case $r = \beta$, by induction on $M \rightarrowtriangle_\beta^* N$. Case $r = \beta\eta$, by the β and η cases. Case $r = h$, by the β case. 3. By induction on $M \rightarrowtriangle_\beta^* N$. □

Normal forms are defined as usual.

Definition 2.9. (Normal forms)
Let $i \in \{1, 2, 3\}$ and $r \in \{\beta, \beta\eta, h\}$.

- $M \in \mathcal{M}_i$ is in r-normal form if there is no $N \in \mathcal{M}_i$ such that $M \rightarrowtriangle_r N$.
- $M \in \mathcal{M}_i$ is r-normalising if there is an $N \in \mathcal{M}_i$ such that $M \rightarrowtriangle_r^* N$ and N is in r-normal.

Finally, the indexed lambda calculi are confluent w.r.t. β-, $\beta\eta$- and h-reductions:

Theorem 2.2. (Confluence)
Let $i \in \{1, 2, 3\}$, $M, M_1, M_2 \in \mathcal{M}_i$, and $r \in \{\beta, \beta\eta, h\}$.

1. If $M \rightarrowtriangle_r^* M_1$ and $M \rightarrowtriangle_r^* M_2$ then there is $M' \in \mathcal{M}_i$ such that $M_1 \rightarrowtriangle_r^* M'$ and $M_2 \rightarrowtriangle_r^* M'$.
2. $M_1 \simeq_r M_2$ iff there is a term $M \in \mathcal{M}_i$ such that $M_1 \rightarrowtriangle_r^* M$ and $M_2 \rightarrowtriangle_r^* M$.

Proof:
We establish the confluence using the parallel reduction method. Full details can be found in the expanded version of this article [22]. □

2.2. The types of the indexed calculi

Let us start by defining type variables and expansion variables.

Definition 2.10. (Type variables and expansion variables)
We assume that a, b range over a countably infinite set of type variables TyVar, and that e ranges over a countably infinite set of expansion variables $\mathsf{ExpVar} = \{\mathsf{e}_0, \mathsf{e}_1, \ldots\}$.

With each expansion variable we associate a unique natural number which is the subscript of the expansion variable. Instead of explicitly naming the elements in ExpVar, we could also have considered a bijective function from expansion variables to natural numbers in order to associate a unique natural number with each expansion variable. We have decided not to do so for clarity reason. Our solution avoids defining an extra function.

For $\lambda I^{\mathbb{N}}$, we study two type systems (none of which has the ω-type). In the first, there are no restrictions on where intersection types and expansion variables occur (see set ITy_1 defined below). In the second, intersections and expansions cannot occur directly to the right of an arrow (see set ITy_2 defined below).

Definition 2.11. (Types, good types and degree of a type for $\lambda I^{\mathbb{N}}$)

- The type set ITy_1 is defined as follows:

$$T, U, V, W \quad \in \quad \mathsf{ITy}_1 \quad ::= \quad a \mid U_1 \to U_2 \mid U_1 \sqcap U_2 \mid eU$$

  The type sets Ty_2 and ITy_2 are defined as follows (note that $\mathsf{Ty}_2 \subseteq \mathsf{ITy}_2 \subseteq \mathsf{ITy}_1$):

$$\begin{array}{lcll} T & \in & \mathsf{Ty}_2 & ::= \quad a \mid U \to T \\ U, V, W & \in & \mathsf{ITy}_2 & ::= \quad U_1 \sqcap U_2 \mid eU \mid T \end{array}$$

- We define a function deg ($\in \mathsf{ITy}_1 \to \mathbb{N}$) by (hence deg is also defined on ITy_2):

$$\begin{array}{lcllcl} \mathsf{deg}(a) & = & 0 & \mathsf{deg}(U \to T) & = & \mathsf{min}(\mathsf{deg}(U), \mathsf{deg}(T)) \\ \mathsf{deg}(eU) & = & \mathsf{deg}(U) + 1 & \mathsf{deg}(U \sqcap V) & = & \mathsf{min}(\mathsf{deg}(U), \mathsf{deg}(V)) \end{array}$$

- We define the set GITy which is the set of good ITy_1 types as follow (this also defines the set of good ITy_2 types: $\mathsf{GITy} \cap \mathsf{ITy}_2$):

$$\begin{array}{lcllcl} a \in \mathsf{TyVar} & & & \Rightarrow & a \in \mathsf{GITy} \\ U \in \mathsf{GITy} & \wedge & e \in \mathsf{ExpVar} & \Rightarrow & eU \in \mathsf{GITy} \\ U, T \in \mathsf{GITy} & \wedge & \mathsf{deg}(U) \geq \mathsf{deg}(T) & \Rightarrow & U \to T \in \mathsf{GITy} \\ U, V \in \mathsf{GITy} & \wedge & \mathsf{deg}(U) = \mathsf{deg}(V) & \Rightarrow & U \sqcap V \in \mathsf{GITy} \end{array}$$

  When $U \in \mathsf{GITy}$, we sometimes say that U is good.

Let $n \leq m$. Let $\vec{\mathsf{e}}_{i(n:m)}U$ or $\vec{\mathsf{e}}_L U$ where $L = (i_n, \ldots, i_m)$ denote $\mathsf{e}_{i_n} \ldots \mathsf{e}_{i_n} U$. Also, let $\vec{e}_{i(n:m),j}U$ denote $e_{\langle n,j \rangle} \ldots e_{\langle m,j \rangle} U$. We consider the application of an expansion variable to a type (eU) to have higher precedence than $\sqcap$ which itself has higher precedence than $\to$. In all our type systems, we quotient types by taking $\sqcap$ to be commutative (i.e., $U_1 \sqcap U_2 = U_2 \sqcap U_1$), associative (i.e., $U_1 \sqcap (U_2 \sqcap U_3) = (U_1 \sqcap U_2) \sqcap U_3$) and idempotent (i.e., $U \sqcap U = U$), by assuming the distributivity of expansion variables over $\sqcap$ (i.e., $e(U_1 \sqcap U_2) = eU_1 \sqcap eU_2$). We denote $U_n \sqcap \ldots \sqcap U_m$ by $\sqcap_{i=n}^{m} U$ (when $n \leq m$).

The next lemma states when arrow, intersection and applications of expansion variables to types are good.

Lemma 2.3. 1. On ITy_1 (hence on ITy_2), we have the following:

   (a) ($U, T \in \mathsf{GITy}$ and $\mathsf{deg}(U) \geq \mathsf{deg}(T)$) iff $U \rightarrow T \in \mathsf{GITy}$.
   (b) ($U, V \in \mathsf{GITy}$ and $\mathsf{deg}(U) = \mathsf{deg}(V)$) iff $U \sqcap V \in \mathsf{GITy}$.
   (c) $U \in \mathsf{GITy}$ iff $eU \in \mathsf{GITy}$.

2. On ITy_2, we have in addition the following:
   (a) If $T \in \mathsf{Ty}_2$ then $\mathsf{deg}(T) = 0$.
   (b) If $\mathsf{deg}(U) = n$ then U is of the form $\sqcap_{i=1}^{m} \vec{e}_{j(1:n),i} V_i$ such that $m \geq 1$ and $\exists i \in \{1, \ldots, m\}$. $V_i \in \mathsf{Ty}_2$.
   (c) If $U \in \mathsf{GITy}$ and $\mathsf{deg}(U) = n$ then U is of the form $\sqcap_{i=1}^{m} \vec{e}_{j(1:n),i} T_i$ such that $m \geq 1$ and $\forall i \in \{1, \ldots, m\}$. $T_i \in \mathsf{Ty}_2 \cap \mathsf{GITy}$.
   (d) $U, T \in \mathsf{GITy}$ iff $U \rightarrow T \in \mathsf{GITy}$ (in ITy_2 and ITy_3).

For $\lambda^{\mathcal{L}_{\mathbb{N}}}$, we study a type system (with the universal type ω). In this type system, in order to get subject reduction and hence completeness, intersections and expansions cannot occur directly to the right of an arrow (see ITy_3 below). Note that the type sets ITy_3 and Ty_3 defined below are far more restricted than the type sets considered for the $\lambda I^{\mathbb{N}}$-calculus and that we do not have the luxury of giving a separate syntax for good types. Note also that the definitions of degrees and types are simultaneous (unlike for ITy_2 and Ty_2 where types were defined without any reference to degrees).

Definition 2.12. (Types and degrees of types for $\lambda^{\mathcal{L}_{\mathbb{N}}}$)

- We define the two sets of types Ty_3 and ITy_3 such that $\mathsf{Ty}_3 \subseteq \mathsf{ITy}_3$, and a function deg ($\in \mathsf{ITy}_3 \rightarrow \mathcal{L}_{\mathbb{N}}$) by simultaneous induction as follows:
  - If $a \in \mathsf{TyVar}$ then $a \in \mathsf{Ty}_3$ and $\mathsf{deg}(a) = \oslash$.
  - If $U \in \mathsf{ITy}_3$ and $T \in \mathsf{Ty}_3$ then $U \rightarrow T \in \mathsf{Ty}_3$ and $\mathsf{deg}(U \rightarrow T) = \oslash$.
  - If $L \in \mathcal{L}_{\mathbb{N}}$ then $\omega^L \in \mathsf{ITy}_3$ and $\mathsf{deg}(\omega^L) = L$.
  - If $U_1, U_2 \in \mathsf{ITy}_3$ and $\mathsf{deg}(U_1) = \mathsf{deg}(U_2)$ then $U_1 \sqcap U_2 \in \mathsf{ITy}_3$ and $\mathsf{deg}(U_1 \sqcap U_2) = \mathsf{deg}(U_1) = \mathsf{deg}(U_2)$.
  - $U \in \mathsf{ITy}_3$ and $\mathsf{e}_i \in \mathsf{ExpVar}$ then $\mathsf{e}_i U \in \mathsf{ITy}_3$ and $\mathsf{deg}(\mathsf{e}_i U) = i :: \mathsf{deg}(U)$.

  Note that deg uses the subscript of expansion variables in order to keep track of the expansion variables contributing to the degree of a type.

- We let T range over Ty_3, and U, V, W range over ITy_3. We quotient types further by having ω^L as a neutral (i.e., $\omega^L \sqcap U = U$). We also assume that for all $i \geq 0$ and $L \in \mathcal{L}_{\mathbb{N}}$, $\mathsf{e}_i \omega^L = \omega^{i::L}$.

All our type systems use the following definition (of course within the corresponding calculus, with the corresponding indices and types):

Definition 2.13. (Environments and typings)

- Let $k \in \{1, 2, 3\}$. We define the three sets of type environments TyEnv_1, TyEnv_2, and TyEnv_3 as follows: $\Gamma, \Delta \in \mathsf{TyEnv}_k = \mathsf{Var}_k \rightarrow \mathsf{ITy}_k$. When writing environments, we sometimes write $x : y$ instead of $x \mapsto y$. We sometimes write $\{x_1^{I_1} \mapsto U_1, \ldots, x_n^{I_n} \mapsto U_n\}$ as $x_1^{I_1} : U_1, \ldots, x_n^{I_n} : U_n$ or as $(x_i^{I_i} : U_i)_n$. We sometimes write $()$ for the empty environment $\varnothing$. If $\mathsf{dj}(\mathsf{dom}(\Gamma_1), \mathsf{dom}(\Gamma_2))$, we write Γ_1, Γ_2 for $\Gamma_1 \cup \Gamma_2$.

- We say that Γ_1 and Γ_2 are joinable and write $\Gamma_1 \diamond \Gamma_2$ iff $(\forall x^{I_1} \in \mathsf{dom}(\Gamma_1).\ x^{I_2} \in \mathsf{dom}(\Gamma_2) \Rightarrow I_1 = I_2)$.
- We say that Γ is OK and write $\mathsf{ok}(\Gamma)$ iff $\forall x^I \in \mathsf{dom}(\Gamma).\ \mathsf{deg}(\Gamma(x^I)) = I$.
- Let $\Gamma_1 = \Gamma_1' \uplus \Gamma_1''$ and $\Gamma_2 = \Gamma_2' \uplus \Gamma_2''$ such that $\mathsf{dj}(\mathsf{dom}(\Gamma_1''), \mathsf{dom}(\Gamma_2''))$, $\mathsf{dom}(\Gamma_1') = \mathsf{dom}(\Gamma_2')$, and $\forall x^I \in \mathsf{dom}(\Gamma_1').\ \mathsf{deg}(\Gamma_1'(x^I)) = \mathsf{deg}(\Gamma_2'(x^I))$. We denote $\Gamma_1 \sqcap \Gamma_2$ the type environment $\{x^I \mapsto \Gamma_1'(x^I) \sqcap \Gamma_2'(x^I) \mid x^I \in \mathsf{dom}(\Gamma_1')\} \cup \Gamma_1'' \cup \Gamma_2''$. Note that $\mathsf{dom}(\Gamma_1 \sqcap \Gamma_2) = \mathsf{dom}(\Gamma_1) \cup \mathsf{dom}(\Gamma_2)$ and that, on environments, $\sqcap$ is commutative, associative and idempotent.
- In $\lambda I^{\mathbb{N}}$ (i.e., on TyEnv_1 and TyEnv_2), we define the set of good type environments as follows: $\mathsf{GTyEnv} = \{\Gamma \mid \forall x^I \in \mathsf{dom}(\Gamma).\ \Gamma(x^I) \in \mathsf{GITy}\}$. If $\Gamma = (x_i^{n_i} : U_i)_m$ then let $\mathsf{deg}(\Gamma) = \min(n_1, \ldots, n_m, \mathsf{deg}(U_1), \ldots, \mathsf{deg}(U_m))$. Let $e\Gamma = \{x^{n+1} \mapsto e\Gamma(x^n) \mid x^n \in \mathsf{dom}(\Gamma)\}$. So $e(\Gamma_1 \sqcap \Gamma_2) = e\Gamma_1 \sqcap e\Gamma_2$.
- In $\lambda^{\mathcal{L}_{\mathbb{N}}}$ (i.e., on TyEnv_3), if $M \in \mathcal{M}_3$ and $\mathsf{fv}(M) = \{x_1^{L_1}, \ldots, x_n^{L_n}\}$ then let $\mathsf{env}_M^{\varnothing}$ be the type environment $(x_i^{L_i} : \omega^{L_i})_n$. For all $\mathsf{e}_j \in \mathsf{ExpVar}$, let $\mathsf{e}_j\Gamma = \{x^{j::L} \mapsto \mathsf{e}_j\Gamma(x^L) \mid x^L \in \mathsf{dom}(\Gamma)\}$. Note that $e(\Gamma_1 \sqcap \Gamma_2) = e\Gamma_1 \sqcap e\Gamma_2$. If $\Gamma = (x_i^{L_i} : U_i)_n$ and $s = \{L \mid \forall i \in \{1, \ldots, n\}.\ L \preceq L_i \wedge L \preceq \mathsf{deg}(U_i)\}$ then $\mathsf{deg}(\Gamma) = L$ such that $L \in s$ and $\forall L' \in s.\ L' \preceq L$.

As we did for terms, we decrease the indexes of types and environments.

Definition 2.14. (Degree decreasing in $\lambda I^{\mathbb{N}}$)

- If $\mathsf{deg}(U) > 0$ then we inductively define the type U^- as follows:

$$(U_1 \sqcap U_2)^- = U_1{}^- \sqcap U_2{}^- \qquad (eU)^- = U$$

  If $\mathsf{deg}(U) \geq n$ then we inductively define the type U^{-n} as follows:

$$U^{-0} = U \qquad U^{-(n+1)} = (U^{-n})^-$$

- If $\mathsf{deg}(\Gamma) > 0$ then let $\Gamma^- = \{x^{n-1} \mapsto \Gamma(x^n)^- \mid x^n \in \mathsf{dom}(\Gamma)\}$.

  If $\mathsf{deg}(\Gamma) \geq n$ then we inductively define the type Γ^{-n} as follows:

$$\Gamma^{-0} = \Gamma \qquad \Gamma^{-(n+1)} = (\Gamma^{-n})^-.$$

Definition 2.15. (Degree decreasing in $\lambda^{\mathcal{L}_{\mathbb{N}}}$)

1. If $\mathsf{deg}(U) \succeq L$ then U^{-L} is inductively defined as follows:

$$U^{-\oslash} = U \qquad (U_1 \sqcap U_2)^{-i::L'} = U_1^{-i::L'} \sqcap U_2^{-i::L'} \qquad (\mathsf{e}_i U)^{-i::L'} = U^{-L'}$$

   We write U^{-i} instead of $U^{-(i)}$.

2. If $\Gamma = (x_i^{L_i} : U_i)_m$ and $\mathsf{deg}(\Gamma) \succeq L$ then by definition $\forall i \in \{1, \ldots, m\}.\ L_i = L :: L_i' \wedge L \preceq \mathsf{deg}(U_i)$, and we define $\Gamma^{-L} = (x^{L_i'} : U_i^{-L})_m$. We write Γ^{-i} instead of $\Gamma^{-(i)}$.

Figure 1 Typing rules / Subtyping rules for $\vdash_1$ and $\vdash_2$

Let $i \in \{1,2\}$. In $\vdash_1$, U and T range over ITy_1. In $\vdash_2$, U ranges over ITy_2 and T ranges only over Ty_2.

$$\frac{T \in \mathsf{GITy} \quad \mathsf{deg}(T) = n}{x^n : \langle (x^n : T) \vdash_1 T\rangle}\ (\mathsf{ax}) \qquad \frac{T \in \mathsf{GITy}}{x^0 : \langle (x^0 : T) \vdash_2 T\rangle}\ (\mathsf{ax}) \qquad \frac{M : \langle \Gamma, (x^n : U) \vdash_i T\rangle}{\lambda x^n.M : \langle \Gamma \vdash_i U \to T\rangle}\ (\to_{\mathsf{I}})$$

$$\frac{M_1 : \langle \Gamma_1 \vdash_i U \to T\rangle \quad M_2 : \langle \Gamma_2 \vdash_i U\rangle \quad \Gamma_1 \diamond \Gamma_2}{M_1 M_2 : \langle \Gamma_1 \sqcap \Gamma_2 \vdash_i T\rangle}\ (\to_{\mathsf{E}}) \qquad \frac{M : \langle \Gamma \vdash_i U\rangle}{M^+ : \langle e\Gamma \vdash_i eU\rangle}\ (\mathsf{exp})$$

$$\frac{M : \langle \Gamma_1 \vdash_i U_1\rangle \quad M : \langle \Gamma_2 \vdash_i U_2\rangle}{M : \langle \Gamma_1 \sqcap \Gamma_2 \vdash_i U_1 \sqcap U_2\rangle}\ (\sqcap_{\mathsf{I}}) \qquad \frac{M : \langle \Gamma \vdash_2 U\rangle \quad \Gamma \vdash_2 U \sqsubseteq \Gamma' \vdash_2 U'}{M : \langle \Gamma' \vdash_2 U'\rangle}\ (\sqsubseteq)$$

The following relation $\sqsubseteq$ is defined on ITy_2, TyEnv_2, and Typing_2:

$$\frac{}{\Psi \sqsubseteq \Psi}\ (\mathsf{ref}) \qquad \frac{\Psi_1 \sqsubseteq \Psi_2 \quad \Psi_2 \sqsubseteq \Psi_3}{\Psi_1 \sqsubseteq \Psi_3}\ (\mathsf{tr}) \qquad \frac{U_2 \in \mathsf{GITy} \quad \mathsf{deg}(U_1) = \mathsf{deg}(U_2)}{U_1 \sqcap U_2 \sqsubseteq U_1}\ (\sqcap_{\mathsf{E}})$$

$$\frac{U_1 \sqsubseteq V_1 \quad U_2 \sqsubseteq V_2}{U_1 \sqcap U_2 \sqsubseteq V_1 \sqcap V_2}\ (\sqcap) \qquad \frac{U_2 \sqsubseteq U_1 \quad T_1 \sqsubseteq T_2}{U_1 \to T_1 \sqsubseteq U_2 \to T_2}\ (\to) \qquad \frac{U_1 \sqsubseteq U_2}{eU_1 \sqsubseteq eU_2}\ (\sqsubseteq_{\mathsf{exp}})$$

$$\frac{U_1 \sqsubseteq U_2 \quad y^n \notin \mathsf{dom}(\Gamma)}{\Gamma, (y^n : U_1) \sqsubseteq \Gamma, (y^n : U_2)}\ (\sqsubseteq_{\mathsf{c}}) \qquad \frac{U_1 \sqsubseteq U_2 \quad \Gamma_2 \sqsubseteq \Gamma_1}{\Gamma_1 \vdash_2 U_1 \sqsubseteq \Gamma_2 \vdash_2 U_2}\ (\sqsubseteq_{\langle\rangle})$$

2.3. The type systems $\vdash_1$ and $\vdash_2$ for $\lambda I^{\mathbb{N}}$ and $\vdash_3$ for $\lambda^{\mathcal{L}_{\mathbb{N}}}$

In this section we introduce our three type systems $\vdash_i$ for $i \in \{1,2,3\}$, our intersection type systems with expansion variables. The system $\vdash_1$ uses the ITy_1 types and the TyEnv_1 type environments, and is for $\lambda I^{\mathbb{N}}$. The system $\vdash_2$ uses the ITy_2 types and the TyEnv_2 type environments, and is for $\lambda I^{\mathbb{N}}$. The system $\vdash_3$ uses the ITy_3 types and the TyEnv_3 type environments, and is for $\lambda^{\mathcal{L}_{\mathbb{N}}}$. In $\vdash_1$, types are not restricted and subject reduction (SR) fails. In $\vdash_2$, the syntax of types is restricted (see ITy_2's definition), and in order to guarantee SR for this type system (and hence completeness later on), we introduce a subtyping relation which allows intersection type elimination (which does not hold in the first type system). In $\vdash_3$, the syntax of types is restricted further (see ITy_3's definition) so that completeness holds with an arbitrary number of expansion variables.

Definition 2.16. (The type systems)

Let $i \in \{1,2,3\}$. The type system $\vdash_i$ uses the set ITy_i of Def. 2.11 (for $i \in \{1,2\}$) and 2.12 (for $i = 3$). The typing rules of $\vdash_1$ and $\vdash_2$ are given on the left of Fig. 1[3]. In $\vdash_1$, U and T range over ITy_1, and Γ range over TyEnv_1. In $\vdash_2$, U range over ITy_2, T range over Ty_2, and Γ range over TyEnv_1. The typing rules of $\vdash_3$ are given on the left of Fig. 2. In both figures, the last clause makes use of a subtyping relation $\sqsubseteq$ which is defined on ITy_2 in Fig. 1 and on ITy_3 in Fig. 2. These subtyping relations are extended to type environments and typings (defined below).

[3]The type system $\vdash_1$ is the smallest relation closed by the rules presented on the left of Fig. 1 (and similarly for $\vdash_2$).

Figure 2 Typing rules / Subtyping rules for $\vdash_3$

U ranges over ITy_3 and T Ty_3.

$$\frac{}{x^{\oslash} : \langle (x^{\oslash} : T) \vdash_3 T\rangle}\ (\mathsf{ax}) \qquad \frac{}{M : \langle \mathsf{env}^{\varnothing}_M \vdash_3 \omega^{\mathsf{deg}(M)}\rangle}\ (\omega)$$

$$\frac{M : \langle \Gamma, (x^L : U) \vdash_3 T\rangle}{\lambda x^L.M : \langle \Gamma \vdash_3 U \to T\rangle}\ (\to_{\mathsf{I}}) \qquad \frac{M : \langle \Gamma \vdash_3 T\rangle \quad x^L \not\in \mathsf{dom}(\Gamma)}{\lambda x^L.M : \langle \Gamma \vdash_3 \omega^L \to T\rangle}\ (\to'_{\mathsf{I}})$$

$$\frac{M_1 : \langle \Gamma_1 \vdash_3 U \to T\rangle \quad M_2 : \langle \Gamma_2 \vdash_3 U\rangle \quad \Gamma_1 \diamond \Gamma_2}{M_1 M_2 : \langle \Gamma_1 \sqcap \Gamma_2 \vdash_3 T\rangle}\ (\to_{\mathsf{E}}) \qquad \frac{M : \langle \Gamma \vdash_3 U\rangle}{M^{+j} : \langle \mathsf{e}_j \Gamma \vdash_3 \mathsf{e}_j U\rangle}\ (\mathsf{exp})$$

$$\frac{M : \langle \Gamma \vdash_3 U_1\rangle \quad M : \langle \Gamma \vdash_3 U_2\rangle}{M : \langle \Gamma \vdash_3 U_1 \sqcap U_2\rangle}\ (\sqcap_{\mathsf{I}}) \qquad \frac{M : \langle \Gamma \vdash_3 U\rangle \quad \Gamma \vdash_3 U \sqsubseteq \Gamma' \vdash_3 U'}{M : \langle \Gamma' \vdash_3 U'\rangle}\ (\sqsubseteq)$$

The following relation $\sqsubseteq$ is defined on ITy_3, TyEnv_3, and Typing_3.

$$\frac{}{\Psi \sqsubseteq \Psi}\ (\mathsf{ref}) \qquad \frac{\Psi_1 \sqsubseteq \Psi_2 \quad \Psi_2 \sqsubseteq \Psi_3}{\Psi_1 \sqsubseteq \Psi_3}\ (\mathsf{tr}) \qquad \frac{\mathsf{deg}(U_1) = \mathsf{deg}(U_2)}{U_1 \sqcap U_2 \sqsubseteq U_1}\ (\sqcap_{\mathsf{E}})$$

$$\frac{U_1 \sqsubseteq V_1 \quad U_2 \sqsubseteq V_2 \quad \mathsf{deg}(U_1) = \mathsf{deg}(U_2)}{U_1 \sqcap U_2 \sqsubseteq V_1 \sqcap V_2}\ (\sqcap) \qquad \frac{U_2 \sqsubseteq U_1 \quad T_1 \sqsubseteq T_2}{U_1 \to T_1 \sqsubseteq U_2 \to T_2}\ (\to)$$

$$\frac{U_1 \sqsubseteq U_2}{eU_1 \sqsubseteq eU_2}\ (\sqsubseteq_{\mathsf{exp}}) \qquad \frac{U_1 \sqsubseteq U_2 \quad y^L \not\in \mathsf{dom}(\Gamma)}{\Gamma, y^L : U_1 \sqsubseteq \Gamma, y^L : U_2}\ (\sqsubseteq_{\mathsf{c}}) \qquad \frac{U_1 \sqsubseteq U_2 \quad \Gamma_2 \sqsubseteq \Gamma_1}{\Gamma_1 \vdash_3 U_1 \sqsubseteq \Gamma_2 \vdash_3 U_2}\ (\sqsubseteq_{\langle\rangle})$$

We define the three typing sets Typing_1, Typing_2, and Typing_3 as follows: $\Phi \in \mathsf{Typing}_i ::= \Gamma \vdash_i U$, where $\Gamma \in \mathsf{TyEnv}_i$ and $U \in \mathsf{ITy}_i$.

Let $\mathsf{Sorts} = \cup_{i=1}^{3}\{\mathsf{Typing}_i, \mathsf{TyEnv}_i, \mathsf{ITy}_i\}$ and let Ψ range over $\cup_{s \in \mathsf{Sorts}} s$.

We say that Γ is $\vdash_i$-legal if there exist M, U such that $M : \langle \Gamma \vdash_i U\rangle$.

Let $j \in \{1, 2\}$. Let $\mathsf{GTyping} = \{\Gamma \vdash_j U \mid \Gamma \in \mathsf{GTyEnv} \wedge U \in \mathsf{GITy}\}$. If $\Phi \in \mathsf{GTyping}$ then we say that Φ is good. Let $\mathsf{deg}(\Gamma \vdash_j U) = \mathsf{min}(\mathsf{deg}(\Gamma), \mathsf{deg}(U))$.

If $s = \{L \mid L \preceq \mathsf{deg}(\Gamma) \wedge L \preceq \mathsf{deg}(U)\}$ then $\mathsf{deg}(\Gamma \vdash_3 U) = L$ such that $L \in s$ and $\forall L' \in s.\ L' \preceq L$.

To illustrate how our indexed type system works, we give an example:

Example 2.1. Let $L_1 = (3) \preceq L_2 = (3, 2) \preceq L_3 = (3, 2, 1) \preceq L_4 = (3, 2, 1, 0)$ and let $a, b, c, d \in \mathsf{TyVar}$. Consider M, M', U as follows:

$$M = \lambda x^{L_2}.\lambda y^{L_1}.(y^{L_1}(x^{L_2}\lambda u^{L_3}.\lambda v^{L_4}.(u^{L_3}(v^{L_4}v^{L_4})))) \in \mathcal{M}_3$$
$$M' = \lambda x^2.\lambda y^1.(y^1(x^2\lambda u^3.\lambda v^4.(u^3(v^4v^4)))) \in \mathcal{M}_2$$
$$U = \mathsf{e}_3(\mathsf{e}_2(\mathsf{e}_1((\mathsf{e}_0 b \to c) \to (\mathsf{e}_0(a \sqcap (a \to b)) \to c)) \to d) \to (((\mathsf{e}_2 d \to a) \sqcap b) \to a)) \in \mathsf{ITy}_2 \cap \mathsf{ITy}_3$$

One can check that $M : \langle () \vdash_3 U\rangle$ and $M' : \langle () \vdash_2 U\rangle$. We simply give some steps in the derivation of $M : \langle () \vdash_3 U\rangle$ (note that the derivation of $M' : \langle () \vdash_2 U\rangle$ only differs from the derivation of $M : \langle () \vdash_3 U\rangle$ by replacing everywhere $\vdash_3$ by $\vdash_2$ and any list $(n_1, \ldots, n_k)$ by k for any $k \geq 0$):

- $v^{\oslash}v^{\oslash} : \langle v^{\oslash} : a \sqcap (a \to b) \vdash_3 b\rangle$
- $v^{(0)}v^{(0)} : \langle v^{(0)} : \mathsf{e}_0(a \sqcap (a \to b)) \vdash_3 \mathsf{e}_0 b\rangle$
- $u^{\oslash} : \langle u^{\oslash} : \mathsf{e}_0 b \to c \vdash_3 \mathsf{e}_0 b \to c\rangle$
- $u^{\oslash}(v^{(0)}v^{(0)}) : \langle u^{\oslash} : \mathsf{e}_0 b \to c, v^{(0)} : \mathsf{e}_0(a \sqcap (a \to b)) \vdash_3 c\rangle$
- $\lambda v^{(0)}.u^{\oslash}(v^{(0)}v^{(0)}) : \langle u^{\oslash} : \mathsf{e}_0 b \to c \vdash_3 \mathsf{e}_0(a \sqcap (a \to b)) \to c\rangle$
- $\lambda u^{\oslash}.\lambda v^{(0)}.u^{\oslash}(v^{(0)}v^{(0)}) : \langle () \vdash_3 (\mathsf{e}_0 b \to c) \to (\mathsf{e}_0(a \sqcap (a \to b)) \to c)\rangle$
- $\lambda u^{(1)}.\lambda v^{(1,0)}.u^{(1)}(v^{(1,0)}v^{(1,0)}) : \langle () \vdash_3 \mathsf{e}_1((\mathsf{e}_0 b \to c) \to (\mathsf{e}_0(a \sqcap (a \to b)) \to c))\rangle$
- $x^{\oslash} : \langle x^{\oslash} : \mathsf{e}_1((\mathsf{e}_0 b \to c) \to (\mathsf{e}_0(a \sqcap (a \to b)) \to c)) \to d \vdash_3 \mathsf{e}_1((\mathsf{e}_0 b \to c) \to (\mathsf{e}_0(a \sqcap (a \to b)) \to c)) \to d\rangle$
- $x^{\oslash}(\lambda u^{(1)}.\lambda v^{(1,0)}.u^{(1)}(v^{(1,0)}v^{(1,0)})) : \langle x^{\oslash} : \mathsf{e}_1((\mathsf{e}_0 b \to c) \to (\mathsf{e}_0(a \sqcap (a \to b)) \to c)) \to d \vdash_3 d\rangle$
- $x^{(2)}(\lambda u^{(2,1)}.\lambda v^{(2,1,0)}.u^{(2,1)}(v^{(2,1,0)}v^{(2,1,0)}))$
  $: \langle x^{(2)} : \mathsf{e}_2(\mathsf{e}_1((\mathsf{e}_0 b \to c) \to (\mathsf{e}_0(a \sqcap (a \to b)) \to c)) \to d) \vdash_3 \mathsf{e}_2 d\rangle$
- $y^{\oslash}(x^{(2)}(\lambda u^{(2,1)}.\lambda v^{(2,1,0)}.u^{(2,1)}(v^{(2,1,0)}v^{(2,1,0)})))$
  $: \langle x^{(2)} : \mathsf{e}_2(\mathsf{e}_1((\mathsf{e}_0 b \to c) \to (\mathsf{e}_0(a \sqcap (a \to b)) \to c)) \to d), y^{\oslash} : (\mathsf{e}_2 d \to a) \sqcap b \vdash_3 a\rangle$
- $\lambda y^{\oslash}.(y^{\oslash}(x^{(2)}(\lambda u^{(2,1)}.\lambda v^{(2,1,0)}.u^{(2,1)}(v^{(2,1,0)}v^{(2,1,0)}))))$
  $: \langle x^{(2)} : \mathsf{e}_2(\mathsf{e}_1((\mathsf{e}_0 b \to c) \to (\mathsf{e}_0(a \sqcap (a \to b)) \to c)) \to d) \vdash_3 ((\mathsf{e}_2 d \to a) \sqcap b) \to a\rangle$
- $\lambda x^{(2)}.\lambda y^{\oslash}.(y^{\oslash}(x^{(2)}(\lambda u^{(2,1)}.\lambda v^{(2,1,0)}.u^{(2,1)}(v^{(2,1,0)}v^{(2,1,0)}))))$
  $: \langle () \vdash_3 \mathsf{e}_2(\mathsf{e}_1((\mathsf{e}_0 b \to c) \to (\mathsf{e}_0(a \sqcap (a \to b)) \to c)) \to d) \to (((\mathsf{e}_2 d \to a) \sqcap b) \to a)\rangle$
- $\lambda x^{L_2}.\lambda y^{L_1}.(y^{L_1}(x^{L_2}(\lambda u^{L_3}.\lambda v^{L_4}.u^{L_3}(v^{L_4}v^{L_4}))))$
  $: \langle () \vdash_3 \mathsf{e}_3(\mathsf{e}_2(\mathsf{e}_1((\mathsf{e}_0 b \to c) \to (\mathsf{e}_0(a \sqcap (a \to b)) \to c)) \to d) \to (((\mathsf{e}_2 d \to a) \sqcap b) \to a))\rangle$

Let us now define our decreasing functions on the Typing_2.

Definition 2.17. 1. If $U \in \mathsf{ITy}_2$ and $\Gamma \in \mathsf{TyEnv}_2$ such that $\mathsf{deg}(\Gamma) > 0$ and $\mathsf{deg}(U) > 0$ then we let $(\Gamma \vdash_2 U)^- = \Gamma^- \vdash_2 U^-$.

2. If $U \in \mathsf{ITy}_3$ and $\Gamma \in \mathsf{TyEnv}_3$ such that $\mathsf{deg}(\Gamma) \succeq L$ and $\mathsf{deg}(U) \succeq L$ then we let $(\Gamma \vdash_3 U)^{-L} = \Gamma^{-L} \vdash_3 U^{-L}$.

Next we show how ordering propagates to environments and relates degrees:

Lemma 2.4. 1. If $\Gamma \sqsubseteq \Gamma'$, $U \sqsubseteq U'$, and $x^I \notin \mathsf{dom}(\Gamma)$ then $\mathsf{dom}(\Gamma) = \mathsf{dom}(\Gamma')$ and $\Gamma, (x^I : U) \sqsubseteq \Gamma', (x^I : U')$.

2. $\Gamma \sqsubseteq \Gamma'$ iff $\Gamma = (x_i^{I_i} : U_i)_n$, $\Gamma' = (x_i^{I_i} : U_i')_n$ and $\forall i \in \{1, \ldots, n\}.\ U_i \sqsubseteq U_i'$.

3. Let $j \in \{2, 3\}$. $\Gamma \vdash_j U \sqsubseteq \Gamma' \vdash_j U'$ iff $\Gamma' \sqsubseteq \Gamma$ and $U \sqsubseteq U'$.

4. If $U_1 \sqsubseteq U_2$ then $\mathsf{deg}(U_1) = \mathsf{deg}(U_2)$ and $U_1 \in \mathsf{GITy} \Leftrightarrow U_2 \in \mathsf{GITy}$.

5. If $\Gamma_1 \sqsubseteq \Gamma_2$ then $\mathsf{deg}(\Gamma_1) = \mathsf{deg}(\Gamma_2)$.

6. Let $j \in \{2, 3\}$. The relation $\sqsubseteq$ is well defined on $\mathsf{ITy}_j \times \mathsf{ITy}_j$, on $\mathsf{TyEnv}_j \times \mathsf{TyEnv}_j$, and on $\mathsf{Typing}_j \times \mathsf{Typing}_j$.

7. If $\Gamma_1, \Gamma_2 \in \mathsf{TyEnv}_2$ and $\Gamma_1 \sqsubseteq \Gamma_2$ then $\Gamma_1 \in \mathsf{GTyEnv} \Leftrightarrow \Gamma_2 \in \mathsf{GTyEnv}$

Proof:
We prove 1. and 2. by induction on the derivation $\Gamma \sqsubseteq \Gamma'$. We prove 3. by induction on the derivation $\Gamma \vdash_j U \sqsubseteq \Gamma' \vdash_j U'$. We prove 4. by induction on the derivation $U_1 \sqsubseteq U_2$. We prove 5. by induction on the derivation $\Gamma_1 \sqsubseteq \Gamma_2$. We prove 6. by induction on a subtyping derivation. We prove 7. by induction on the derivation of $\Gamma_1 \sqsubseteq \Gamma_2$. □

The next theorem states that typings are well defined, that within a typing, degrees are well behaved and that we do not allow weakening.

Theorem 2.3. Let $j \in \{1, 2, 3\}$. We have:

1. $\vdash_j$ is well defined on $\mathcal{M}_j \times \mathsf{TyEnv}_j \times \mathsf{ITy}_j$.

2. Let $M : \langle \Gamma \vdash_j U \rangle$.

   (a) $\mathsf{deg}(M) = \mathsf{deg}(U)$, $\mathsf{ok}(\Gamma)$, and $\mathsf{dom}(\Gamma) = \mathsf{fv}(M)$.
   (b) If $j \neq 3$ then $U \in \mathsf{GITy}$, $M \in \mathbb{M}$, $\Gamma \in \mathsf{GTyEnv}$, and $\mathsf{deg}(\Gamma) \geq \mathsf{deg}(M)$.
   (c) If $j = 3$ then $\mathsf{deg}(\Gamma) \succeq \mathsf{deg}(U)$.
   (d) If $j = 2$ and $\mathsf{deg}(U) \geq k$ then $M^{-k} : \langle \Gamma^{-k} \vdash_2 U^{-k} \rangle$.
   (e) If $j = 3$ and $\mathsf{deg}(U) \succeq K$ then $M^{-K} : \langle \Gamma^{-K} \vdash_3 U^{-K} \rangle$.

Proof:
We prove 1. and 2. by induction on the derivation $M : \langle \Gamma \vdash_j U \rangle$. □

Let us now present admissible typing (and subtyping) rules.

Remark 2.1. 1. The rule $\dfrac{M : \langle \Gamma_1 \vdash_3 U_1 \rangle \quad M : \langle \Gamma_2 \vdash_3 U_2 \rangle}{M : \langle \Gamma_1 \sqcap \Gamma_2 \vdash_3 U_1 \sqcap U_2 \rangle}\ (\sqcap_{\mathsf{I}}')$ is admissible

2. The rule $\dfrac{U \in \mathsf{GITy} \quad \mathsf{deg}(U) = n}{x^n : \langle (x^n : U) \vdash_2 U \rangle}\ (\mathsf{ax}')$ is admissible

3. The rule $\dfrac{}{x^{\mathsf{deg}(U)} : \langle (x^{\mathsf{deg}(U)} : U) \vdash_3 U \rangle}\ (\mathsf{ax}'')$ is admissible

4. The rule $\dfrac{}{U \sqsubseteq \omega^{\mathsf{deg}(U)}}\ (\omega')$ is admissible

Let us now present some results concerning the ω type and joinability.

Lemma 2.5. 1. If $M : \langle \Gamma \vdash_3 U \rangle$ then $\Gamma \sqsubseteq \mathsf{env}^{\varnothing}_M$

2. If $\mathsf{dom}(\Gamma) = \mathsf{fv}(M)$ and $\mathsf{ok}(\Gamma)$ then $M : \langle \Gamma \vdash_3 \omega^{\mathsf{deg}(M)} \rangle$.

3. If $i \in \{1, 2, 3\}$, $M_1 : \langle \Gamma_1 \vdash_i U_1 \rangle$ and $M_2 : \langle \Gamma_2 \vdash_i U_2 \rangle$ then $\Gamma_1 \diamond \Gamma_2 \Leftrightarrow M_1 \diamond M_2$.

Proof:

1. Let $\Gamma = (x_i^{L_i} : U_i)_n$ where $\mathsf{fv}(M) = \{x_1^{L_1}, \ldots, x_n^{L_n}\}$ by Theorem 2.3.2a. By Remark 2.1.4, $\forall i \in \{1, \ldots, n\}.\ U_i \sqsubseteq \omega^{\mathsf{deg}(U_i)}$. By Theorem 2.3.2a, $\mathsf{ok}(\Gamma)$ and therefore $\forall i \in \{1, \ldots, n\}.\ \mathsf{deg}(U_i) = L_i$. Finally, by Lemma 2.4.2, $\Gamma \sqsubseteq \mathsf{env}^{\varnothing}_M$.

2. Let $\Gamma = (x_i^{L_i} : U_i)_n$. Then by hypotheses $\mathsf{fv}(M) = \{x_1^{L_1}, \ldots, x_n^{L_n}\}$ and $\forall i \in \{1, \ldots, n\}.\ \mathsf{deg}(U_i) = L_i$. By Remark 2.1.4, $\forall i \in \{1, \ldots, n\}.\ U_i \sqsubseteq \omega^{L_i}$. By Lemma 2.4.2, $\Gamma \sqsubseteq \mathsf{env}^{\varnothing}_M = (x^{L_i} : \omega^{L_i})_n$. Since by rule (ω), $M : \langle \mathsf{env}^{\varnothing}_M \vdash_3 \omega^{\mathsf{deg}(M)} \rangle$, we have by rules $(\sqsubseteq)$ and $(\sqsubseteq_{\langle\rangle})$, $M : \langle \Gamma \vdash_3 \omega^{\mathsf{deg}(M)} \rangle$.

3. $\Leftarrow$) Let $x^{I_1} \in \mathsf{dom}(\Gamma_1)$ and $x^{I_2} \in \mathsf{dom}(\Gamma_2)$ then by Theorem 2.3.2a, $x^{I_1} \in \mathsf{fv}(M_1)$ and $x^{I_2} \in \mathsf{fv}(M_2)$. Because $M_1 \diamond M_2$, then $I_1 = I_2$ and therefore $\Gamma_1 \diamond \Gamma_2$. $\Rightarrow$) Let $x^{I_1} \in \mathsf{fv}(M_1)$ and $x^{I_2} \in \mathsf{fv}(M_2)$ then by Theorem 2.3.2a, $x^{I_1} \in \mathsf{dom}(\Gamma_1)$ and $x^{I_2} \in \mathsf{dom}(\Gamma_2)$. Because $\Gamma_1 \diamond \Gamma_2$, then $I_1 = I_2$ and therefore $M_1 \diamond M_2$.

$\square$

2.4. Subject reduction and expansion properties of our type systems

2.4.1. Subject reduction and expansion properties for $\vdash_1$ and $\vdash_2$

Now we list the generation lemmas for $\vdash_1$ and $\vdash_2$ (for proofs see the expanded version of this article [22]).

Lemma 2.6. (Generation for $\vdash_1$)

1. If $x^n : \langle \Gamma \vdash_1 T \rangle$ then $\Gamma = (x^n : T)$.

2. If $\lambda x^n.M : \langle \Gamma \vdash_1 T_1 \to T_2 \rangle$ then $M : \langle \Gamma, x^n : T_1 \vdash_1 T_2 \rangle$.

3. If $MN : \langle \Gamma \vdash_1 T \rangle$ and $\mathsf{deg}(T) = m$ then $\Gamma = \Gamma_1 \sqcap \Gamma_2$, $T = \sqcap_{i=1}^{n} \vec{e}_{j(1:m),i} T_i$, $n \geq 1$, $M : \langle \Gamma_1 \vdash_1 \sqcap_{i=1}^{n} \vec{e}_{j(1:m),i}(T'_i \to T_i) \rangle$ and $N : \langle \Gamma_2 \vdash_1 \sqcap_{i=1}^{n} \vec{e}_{j(1:m),i} T'_i \rangle$.

Lemma 2.7. (Generation for $\vdash_2$)

1. If $x^n : \langle \Gamma \vdash_2 U \rangle$ then $\Gamma = (x^n : V)$ where $V \sqsubseteq U$.

2. If $\lambda x^n.M : \langle \Gamma \vdash_2 U \rangle$ and $\mathsf{deg}(U) = m$ then $U = \sqcap_{i=1}^{k} \vec{e}_{j(1:m),i}(V_i \to T_i)$ where $k \geq 1$ and $\forall i \in \{1, \ldots, k\}.\ M : \langle \Gamma, x^n : \vec{e}_{j(1:m),i} V_i \vdash_2 \vec{e}_{j(1:m),i} T_i \rangle$.

3. If $MN : \langle \Gamma \vdash_2 U \rangle$ and $\mathsf{deg}(U) = m$ then $U = \sqcap_{i=1}^{k} \vec{e}_{j(1:m),i} T_i$ where $k \geq 1$, $\Gamma = \Gamma_1 \sqcap \Gamma_2$, $M : \langle \Gamma_1 \vdash_2 \sqcap_{i=1}^{k} \vec{e}_{j(1:m),i}(U_i \to T_i) \rangle$, and $N : \langle \Gamma_2 \vdash_2 \sqcap_{i=1}^{k} \vec{e}_{j(1:m),i} U_i \rangle$.

We also show that no β-redexes are blocked in a typable term.

Remark 2.2. (No β-redexes are blocked in typable terms)
Let $i \in \{1, 2\}$ and $M : \langle \Gamma \vdash_i U \rangle$. If $(\lambda x^n.M_1)M_2$ is a subterm of M then $\mathsf{deg}(M_2) = n$ and hence $(\lambda x^n.M_1)M_2 \rightarrow\!\!\!\!\triangleright_\beta M_1[x^n := M_2]$.

Lemma 2.8. (Substitution for $\vdash_2$)
If $M : \langle \Gamma, x^I : U \vdash_2 V \rangle$, $N : \langle \Delta \vdash_2 U \rangle$ and $M \diamond N$ then $M[x^I := N] : \langle \Gamma \sqcap \Delta \vdash_2 V \rangle$.

Proof:
By induction on the derivation $M : \langle \Gamma, x^I : U \vdash_2 V \rangle$. □

Lemma 2.9. (Substitution and Subject β-reduction fails for $\vdash_1$)
Let a, b, c be different type variables. We have:

1. $(\lambda x^0.x^0 x^0)(y^0 z^0) \rightarrowtriangle_\beta (y^0 z^0)(y^0 z^0)$.
2. $x^0 x^0 : \langle x^0 : (a \to c) \sqcap a \vdash_1 c \rangle$.
3. $(\lambda x^0.x^0 x^0)(y^0 z^0) : \langle y^0 : b \to ((a \to c) \sqcap a), z^0 : b \vdash_1 c \rangle$.
4. It is not possible that $(y^0 z^0)(y^0 z^0) : \langle y^0 : b \to ((a \to c) \sqcap a), z^0 : b \vdash_1 c \rangle$.

Hence, the substitution and subject β-reduction lemmas fail for $\vdash_1$.

Proof:
1., 2., and 3. are easy.

For 4., assume $(y^0 z^0)(y^0 z^0) : \langle y^0 : b \to ((a \to c) \sqcap a), z^0 : b \vdash_1 c \rangle$. By Lemma 2.6.3 twice, Theorem 2.3 and Lemma 2.6.1:

- $y^0 z^0 : \langle y^0 : b \to ((a \to c) \sqcap a), z^0 : b \vdash_1 \sqcap_{i=1}^n (T_i \to c) \rangle$ and $n \geq 1$.
- $y^0 : \langle y^0 : b \to ((a \to c) \sqcap a) \vdash_1 \sqcap_{i=1}^n T'_i \to T_i \to c \rangle$.
- $\sqcap_{i=1}^n T'_i \to T_i \to c = b \to ((a \to c) \sqcap a)$.

Hence, for some $i \in \{1, \ldots, n\}$, $b = T'_i$ and $T_i \to c = (a \to c) \sqcap a$ which is absurd. □

Nevertheless, we show that β subject reduction and expansion hold in $\vdash_2$. This will be used in the proof of completeness (more specifically in Lemma 3.6 which is the basis of the completeness Theorem 3.1).

Lemma 2.10. (Subject reduction and expansion for $\vdash_2$ w.r.t. β)

1. If $M : \langle \Gamma \vdash_2 U \rangle$ and $M \rightarrowtriangle_\beta^* N$ then $N : \langle \Gamma \vdash_2 U \rangle$.
2. If $N : \langle \Gamma \vdash_2 U \rangle$ and $M \rightarrowtriangle_\beta^* N$ then $M : \langle \Gamma \vdash_2 U \rangle$.

2.4.2. Subject reduction and expansion properties for $\vdash_3$

Now we list the generation lemmas for $\vdash_3$ (for proofs see the expanded version of this article [22]).

Lemma 2.11. (Generation for $\vdash_3$)

1. If $x^L : \langle \Gamma \vdash_3 U \rangle$ then $\Gamma = (x^L : V)$ and $V \sqsubseteq U$.
2. If $\lambda x^L.M : \langle \Gamma \vdash_3 U \rangle$, $x^L \in \mathsf{fv}(M)$ and $\mathsf{deg}(U) = K$ then $U = \omega^K$ or $U = \sqcap_{i=1}^p \vec{\mathsf{e}}_K (V_i \to T_i)$ where $p \geq 1$ and $\forall i \in \{1, \ldots, p\}$. $M : \langle \Gamma, x^L : \vec{\mathsf{e}}_K V_i \vdash_3 \vec{\mathsf{e}}_K T_i \rangle$.

3. If $\lambda x^L.M : \langle \Gamma \vdash_3 U \rangle$, $x^L \notin \mathsf{fv}(M)$ and $\mathsf{deg}(U) = K$ then $U = \omega^K$ or $U = \sqcap_{i=1}^{p} \vec{\mathsf{e}}_K(V_i \to T_i)$ where $p \geq 1$ and $\forall i \in \{1, \ldots, p\}$. $M : \langle \Gamma \vdash_3 \vec{\mathsf{e}}_K T_i \rangle$.
4. If $Mx^L : \langle \Gamma, (x^L : U) \vdash_3 T \rangle$ and $x^L \notin \mathsf{fv}(M)$, then $M : \langle \Gamma \vdash_3 U \to T \rangle$.

Proof:
1. By induction on the derivation $x^L : \langle \Gamma \vdash_3 U \rangle$. 2. By induction on the derivation $\lambda x^L.M : \langle \Gamma \vdash_3 U \rangle$. 3. Same proof as that of 2. 4. By induction on the derivation $Mx^L : \langle \Gamma, x^L : U \vdash_3 T \rangle$. □

Lemma 2.12. (Substitution for $\vdash_3$)
If $M : \langle \Gamma, x^L : U \vdash_3 V \rangle$, $N : \langle \Delta \vdash_3 U \rangle$ and $M \diamond N$ then $M[x^L := N] : \langle \Gamma \sqcap \Delta \vdash_3 V \rangle$.

Proof:
By induction on the derivation $M : \langle \Gamma, x^L : U \vdash_3 V \rangle$. □

Since $\vdash_3$ does not allow weakening, we need the next definition since when a term is reduced, it may lose some of its free variables and hence will need to be typed in a smaller environment.

Definition 2.18. Let $\Gamma\restriction_s$ stand for $s \lhd \Gamma$. We write $\Gamma\restriction_M$ instead of $\Gamma\restriction_{\mathsf{fv}(M)}$.

Now we are ready to prove the main result of this section:

Theorem 2.4. (Subject reduction for $\vdash_3$)
If $M : \langle \Gamma \vdash_3 U \rangle$ and $M \rightarrow\!\!\!\!\triangleright^*_{\beta\eta} N$ then $N : \langle \Gamma\restriction_N \vdash_3 U \rangle$.

Proof:
By induction on the reduction $M \rightarrow\!\!\!\!\triangleright^*_{\beta\eta} N$. □

Corollary 2.1. 1. If $M : \langle \Gamma \vdash_3 U \rangle$ and $M \rightarrow\!\!\!\!\triangleright^*_{\beta} N$ then $N : \langle \Gamma\restriction_N \vdash_3 U \rangle$.

2. If $M : \langle \Gamma \vdash_3 U \rangle$ and $M \rightarrow\!\!\!\!\triangleright^*_{h} N$ then $N : \langle \Gamma\restriction_N \vdash_3 U \rangle$.

The next lemma is needed for expansion.

Lemma 2.13. If $M[x^L := N] : \langle \Gamma \vdash_3 U \rangle$, $\mathsf{deg}(N) = L$, $x^L \in \mathsf{fv}(M)$, and $M \diamond N$ then there exist a type V and two type environments Γ_1, Γ_2 such that $\mathsf{deg}(V) = L$, $M : \langle \Gamma_1, x^L : V \vdash_3 U \rangle$, $N : \langle \Gamma_2 \vdash_3 V \rangle$, and $\Gamma = \Gamma_1 \sqcap \Gamma_2$.

Proof:
By induction on the derivation $M[x^L := N] : \langle \Gamma \vdash_3 U \rangle$. □

Since more free variables might appear in the β-expansion of a term, the next definition gives a possible enlargement of an environment.

Definition 2.19. Let $m \geq n$, $\Gamma = (x_i^{L_i} : U_i)_n$ and $X = \{x_1^{L_1}, \ldots, x_m^{L_m}\}$. We write $\Gamma\uparrow^X$ for $x_1^{L_1} : U_1, \ldots, x_n^{L_n} : U_n, x_{n+1}^{L_{n+1}} : \omega^{L_{n+1}}, \ldots, x_m^{L_m} : \omega^{L_m}$. If $\mathsf{dom}(\Gamma) \subseteq \mathsf{fv}(M)$, we write $\Gamma\uparrow^M$ instead of $\Gamma\uparrow^{\mathsf{fv}(M)}$.

We are now ready to establish that subject β-expansion holds in $\vdash_3$ (Theorem. 2.5) and that subject η-expansion fails (Lemma 2.14).

Theorem 2.5. (Subject β-expansion holds in $\vdash_3$)
If $N : \langle \Gamma \vdash_3 U \rangle$ and $M \rightarrow\!\!\!\!\triangleright^*_\beta N$ then $M : \langle \Gamma \uparrow^M \vdash_3 U \rangle$.

Proof:
By induction on the length of the derivation $M \rightarrow\!\!\!\!\triangleright^*_\beta N$ using the fact that if $\mathsf{fv}(P) \subseteq \mathsf{fv}(Q)$ then $(\Gamma \uparrow^P)\uparrow^Q = \Gamma \uparrow^Q$. □

Corollary 2.2. If $N : \langle \Gamma \vdash_3 U \rangle$ and $M \rightarrow\!\!\!\!\triangleright^*_h N$ then $M : \langle \Gamma \uparrow^M \vdash_3 U \rangle$.

Lemma 2.14. (Subject η-expansion fails in $\vdash_3$)
Let a be a type variable and let $x \neq y$. We have:

1. $\lambda y^\oslash.\lambda x^\oslash.y^\oslash x^\oslash \rightarrow\!\!\!\!\triangleright_\eta \lambda y^\oslash.y^\oslash$.
2. $\lambda y^\oslash.y^\oslash : \langle () \vdash_3 a \to a \rangle$.
3. It is not possible that: $\lambda y^\oslash.\lambda x^\oslash.y^\oslash x^\oslash : \langle () \vdash_3 a \to a \rangle$. Hence, subject η-expansion fails in $\vdash_3$.

Proof:
1. and 2. are easy. For 3., assume $\lambda y^\oslash.\lambda x^\oslash.y^\oslash x^\oslash : \langle () \vdash_3 a \to a \rangle$. By Lemma 2.11.2, $\lambda x^\oslash.y^\oslash x^\oslash : \langle (y : a) \vdash_3 a \rangle$. Again, by Lemma 2.11.2, $a = \omega^\oslash$ or there exists $n \geq 1$ such that $a = \sqcap_{i=1}^n (U_i \to T_i)$, absurd. □

3. Realisability semantics and their completeness

3.1. Realisability

Crucial to a realisability semantics is the notion of a saturated set:

Definition 3.1. (Saturated sets)
Let $i \in \{1, 2, 3\}$ and $\overline{M}, \overline{M}_1, \overline{M}_2 \subseteq \mathcal{M}_i$.

1. Let $\overline{M}_1 \rightsquigarrow \overline{M}_2 = \{M \in \mathcal{M}_i \mid \forall N \in \overline{M}_1.\ M \diamond N \Rightarrow MN \in \overline{M}_2\}$.
2. Let $\overline{M}_1 \wr \overline{M}_2$ iff $\forall M \in \overline{M}_1 \rightsquigarrow \overline{M}_2.\ \exists N \in \overline{M}_1.\ M \diamond N$.
3. For $r \in \{\beta, \beta\eta, h\}$, let $\mathsf{SAT}^r = \{\overline{M} \subseteq \mathcal{M}_i \mid (M \rightarrow\!\!\!\!\triangleright^*_r N \wedge N \in \overline{M}) \Rightarrow M \in \overline{M}\}$. If $\overline{M} \in \mathsf{SAT}^r$ then we say that $\overline{M}$ is r-saturated.

Saturation is closed under intersection, lifting and arrows:

Lemma 3.1. Let $i \in \{1, 2, 3\}$, $r \in \{\beta, \beta\eta, h\}$, and $\overline{M}_1, \overline{M}_2 \subseteq \mathcal{M}_i$.

1. If $\overline{M}_1, \overline{M}_2$ are r-saturated sets then $\overline{M}_1 \cap \overline{M}_2$ is r-saturated.
2. If $\overline{M}_1 \subseteq \mathcal{M}_2$ is r-saturated then $\overline{M}_1{}^+$ is r-saturated.

3. If $\overline{M}_1 \subseteq \mathcal{M}_3$ is r-saturated then $\overline{M}_1^{+i}$ is r-saturated.

4. If $\overline{M}_2$ is r-saturated then $\overline{M}_1 \rightsquigarrow \overline{M}_2$ is r-saturated.

5. If $\overline{M}_1, \overline{M}_2 \subseteq \mathcal{M}_2$ then $(\overline{M}_1 \rightsquigarrow \overline{M}_2)^+ \subseteq \overline{M}_1{}^+ \rightsquigarrow \overline{M}_2{}^+$.

6. If $\overline{M}_1, \overline{M}_2 \subseteq \mathcal{M}_3$ then $(\overline{M}_1 \rightsquigarrow \overline{M})^{+i} \subseteq \overline{M}_1^{+i} \rightsquigarrow \overline{M}_2^{+i}$.

7. Let $\overline{M}_1, \overline{M}_2 \subseteq \mathcal{M}_2$. If $\overline{M}_1{}^+ \wr \overline{M}_2{}^+$, then $\overline{M}_1{}^+ \rightsquigarrow \overline{M}_2{}^+ \subseteq (\overline{M}_1 \rightsquigarrow \overline{M}_2)^+$.

8. Let $\overline{M}_1, \overline{M}_2 \subseteq \mathcal{M}_3$. If $\overline{M}_1^{+i} \wr \overline{M}_2^{+i}$, then $\overline{M}_1^{+i} \rightsquigarrow \overline{M}_2^{+i} \subseteq (\overline{M}_1 \rightsquigarrow \overline{M}_2)^{+i}$.

9. For every $n \in \mathbb{N}$, the set $\mathbb{M}^n$ is r-saturated.

The interpretations and meanings of types are crucial to a realisability semantics:

Definition 3.2. (Interpretations and meaning of types)
Let $\mathsf{Var} = \mathsf{Var}_1 \cup \mathsf{Var}_2$ such that $\mathsf{dj}(\mathsf{Var}_1, \mathsf{Var}_2)$ and $\mathsf{Var}_1, \mathsf{Var}_2$ are both countably infinite. Let $i \in \{1, 2, 3\}$.

1. Let $x \in \mathsf{Var}_i$ and I an index. We define the following family of sets:

$$\mathsf{VAR}_x^I = \{M \in \mathcal{M}_i \mid \exists N_1, \ldots, N_n \in \mathcal{M}_i.\ M = x^I N_1 \ldots N_n\}.$$

2. In $\lambda I^{\mathbb{N}}$, let $r = \beta$ and $I_0 = 0$. In $\lambda^{\mathcal{L}_{\mathbb{N}}}$, let $r \in \{\beta, \beta\eta, h\}$ and $I_0 = \oslash$.

   (a) An r_i-interpretation $\mathcal{I}$ is a function in $\mathsf{TyVar} \to \mathbb{P}(\mathcal{M}_i^{I_0})$ such that for all $a \in \mathsf{TyVar}$:

   $$\mathcal{I}(a) \in \mathsf{SAT}^r \qquad \forall x \in \mathsf{Var}_1.\ \mathsf{VAR}_x^{I_0} \subseteq \mathcal{I}(a) \qquad \text{In } \lambda I^{\mathbb{N}}, \mathcal{I}(a) \subseteq \mathbb{M}^0$$

   (b) We extend $\mathcal{I}$ to ITy_1 in case of $\lambda I^{\mathbb{N}}$ and to ITy_3 in case of $\lambda^{\mathcal{L}_{\mathbb{N}}}$ as follows:

In $\lambda I^{\mathbb{N}}$ and $\lambda^{\mathcal{L}_{\mathbb{N}}}$:	$\mathcal{I}(U_1 \sqcap U_2) = \mathcal{I}(U_1) \cap \mathcal{I}(U_2)$	$\mathcal{I}(U \to T) = \mathcal{I}(U) \rightsquigarrow \mathcal{I}(T)$
In $\lambda I^{\mathbb{N}}$:	$\mathcal{I}(eU) = \mathcal{I}(U)^+$	
In $\lambda^{\mathcal{L}_{\mathbb{N}}}$:	$\mathcal{I}(\mathsf{e}_i U) = \mathcal{I}(U)^{+i}$	$\mathcal{I}(\omega^L) = \mathcal{M}_3^L$

   Let $\mathsf{Interp}^{r_i} = \{\mathcal{I} \mid \mathcal{I} \text{ is a } r_i\text{-interpretation}\}$[4].

   (c) Let $U \in \mathsf{ITy}_i$. We define $[U]_{r_i}$, the r_i-interpretation of U as follows:

   $$[U]_{r_i} = \{M \in \mathcal{M}_i \mid \mathsf{closed}(M) \wedge M \in \textstyle\bigcap_{\mathcal{I} \in \mathsf{Interp}^{r_i}} \mathcal{I}(U)\}$$

Because $\cap$ is commutative, associative, idempotent, $(\overline{M}_1 \cap \overline{M}_2)^+ = \overline{M}_1{}^+ \cap \overline{M}_2{}^+$ in $\lambda I^{\mathbb{N}}$, $(\overline{M}_1 \cap \overline{M}_2)^{+i} = \overline{M}_1^{+i} \cap \overline{M}_2^{+i}$ in $\lambda^{\mathcal{L}_{\mathbb{N}}}$, and $\mathcal{I}$ is well defined.

Type interpretations are saturated and interpretations of good types contain only good terms.

Lemma 3.2. Let $r \in \{\beta, \beta\eta, h\}$. Let $i \in \{1, 2, 3\}$.

1. (a) For all $U \in \mathsf{ITy}_i$ and $\mathcal{I} \in \mathsf{Interp}^{r_i}$, we have $\mathcal{I}(U) \in \mathsf{SAT}^r$.

[4] We effectively define five interpretation sets $\mathsf{Interp}^{\beta_1}$, $\mathsf{Interp}^{\beta_2}$, $\mathsf{Interp}^{\beta_3}$, $\mathsf{Interp}^{\beta\eta_3}$, and Interp^{h_3}

(b) If $\mathsf{deg}(U) = L$ and $\mathcal{I} \in \mathsf{Interp}^{r_3}$ then $\forall x \in \mathsf{Var}_1.\ \mathsf{VAR}^L_x \subseteq \mathcal{I}(U) \subseteq \mathcal{M}^L_3$.

(c) On ITy_1 (hence also on ITy_2), if $U \in \mathsf{GITy}$, $\mathsf{deg}(U) = n$, and $\mathcal{I} \in \mathsf{Interp}^{r_2}$ then $\forall x \in \mathsf{Var}_1.\ x^n \in \mathsf{VAR}^n_x \subseteq \mathcal{I}(U) \subseteq \mathbb{M}^n$.

2. Let $i \in \{2, 3\}$. If $\mathcal{I} \in \mathsf{Interp}^{r_i}$ and $U \sqsubseteq V$ then $\mathcal{I}(U) \subseteq \mathcal{I}(V)$.

Proof:
1a . By induction on U using Lemma 3.1. 1b. By induction on U. 1c. By definition, $x^n \in \mathsf{VAR}^n_x$. We prove $\mathsf{VAR}^n_x \subseteq \mathcal{I}(U) \subseteq \mathbb{M}^n$ by induction on $U \in \mathsf{GITy}$. 2. By induction of the derivation $U \sqsubseteq V$. $\square$

Corollary 3.1. (Meanings of good types consist of good terms)
On ITy_1 (hence also on ITy_2), if $U \in \mathsf{GITy}$ such that $\mathsf{deg}(U) = n$ then $[U]_{\beta_2} \subseteq \mathbb{M}^n$.

Proof:
By Lemma 3.2.1c, for any interpretation $\mathcal{I} \in \mathsf{Interp}^{\beta_2}$, $\mathcal{I}(U) \subseteq \mathbb{M}^n$. $\square$

Lemma 3.3. (Soundness of $\vdash_1$, $\vdash_2$, and $\vdash_3$)
Let $i \in \{1, 2, 3\}$, $r \in \{\beta, \beta\eta, h\}$, $\mathcal{I} \in \mathsf{Interp}^{r_i}$. If $M : \langle (x_j^{I_j} : U_j)_n \vdash_i U \rangle$, $\forall j \in \{1, \ldots, n\}.\ N_j \in \mathcal{I}(U_j)$, and $\diamond\{M, N_1, \ldots, N_n\}$ then $M[(x_j^{I_j} := N_j)_n] \in \mathcal{I}(U)$.

Proof:
By induction on the derivation $M : \langle (x_j^{I_j} : U_j)_n \vdash_i U \rangle$. $\square$

Corollary 3.2. Let $r \in \{\beta, \beta\eta, h\}$ and $i \in \{1, 2, 3\}$. If $M : \langle () \vdash_i U \rangle$ then $M \in [U]_{r_i}$.

Proof:
By Lemma 3.3, $M \in \mathcal{I}(U)$ for any $\mathcal{I} \in \mathsf{Interp}^{r_i}$. By Theorem 2.3, $\mathsf{fv}(M) = \mathsf{dom}(()) = \varnothing$ and hence M is closed. Therefore, $M \in [U]_{r_i}$. $\square$

Lemma 3.4. (The meaning of types is closed under type operations)
Let $r \in \{\beta, \beta\eta, h\}$ and $j \in \{1, 2, 3\}$. The following hold:

1. $[\mathsf{e}_i U]_{r_3} = [U]^{+i}_{r_3}$ and if $j \neq 3$ then $[eU]_{r_j} = [U]_{r_j}{}^{+}$.

2. $[U \sqcap V]_{r_j} = [U]_{r_j} \cap [V]_{r_j}$.

3. If $U \to T \in \mathsf{ITy}_3$ then $\forall \mathcal{I} \in \mathsf{Interp}^{r_3}.\ \mathcal{I}(U) \wr \mathcal{I}(T)$.

4. If $U \to T \in \mathsf{GITy}$ then $\forall \mathcal{I} \in \mathsf{Interp}^{\beta_2}.\ \mathcal{I}(U) \wr \mathcal{I}(T)$.

5. On ITy_1 only (since $eU \to eT \notin \mathsf{ITy}_2$), we have: if $U \to T \in \mathsf{GITy}$ then $[e(U \to T)]_{\beta_2} = [eU \to eT]_{\beta_2}$.

Proof:
1. and 2. are easy.

3. Let $\mathsf{deg}(U) = L$, $M \in \mathcal{I}(U) \rightsquigarrow \mathcal{I}(T)$ and $x \in \mathsf{Var}_1$ such that $\forall K.\ x^K \notin \mathsf{fv}(M)$, hence $M \diamond x^L$ and by Lemma 3.2, $x^L \in \mathcal{I}(U)$.

4. Let $\mathsf{deg}(U) = n$ and $M \in \mathcal{I}(U) \rightsquigarrow \mathcal{I}(T)$. Take $x \in \mathsf{Var}_1$ such that $\forall p.\ x^p \notin \mathsf{fv}(M)$. Hence, $M \diamond x^n$. By Lemma 2.3, $U \in \mathsf{GITy}$ and by Lemma 3.2, $x^n \in \mathcal{I}(U)$.

5. Since $U \to T \in \mathsf{GITy}$ then, by Lemma 2.3, $U, T \in \mathsf{GITy}$ and $\mathsf{deg}(U) \geq \mathsf{deg}(T)$. Again by Lemma 2.3, $eU, eT \in \mathsf{GITy}$, $\mathsf{deg}(eU) \geq \mathsf{deg}(eT)$ and $eU \to eT \in \mathsf{GITy}$. Hence by 4., $\mathcal{I}(U)^+ \wr \mathcal{I}(T)^+$. Thus, by Lemma 3.1.5 and Lemma 3.1.7, $\forall \mathcal{I} \in \mathsf{Interp}^{\beta_2}.\ \mathcal{I}(e(U \to T)) = \mathcal{I}(eU \to eT)$.

$\square$

Let us now put the realisability semantics in use.

Example 3.1. Let $\mathtt{a}$ and $\mathtt{b}$ be two distinct type variables in TyVar. We define:

- $\mathsf{id}_0 = \mathtt{a} \to \mathtt{a}$ and $\mathtt{id}_1 = \mathtt{e}_1(\mathsf{id}_0)$.
- $\mathsf{d} = (\mathtt{a} \sqcap (\mathtt{a} \to \mathtt{b})) \to \mathtt{b}$.
- $\mathsf{nat}_0 = (\mathtt{a} \to \mathtt{a}) \to (\mathtt{a} \to \mathtt{a})$, $\mathtt{nat}_1 = \mathtt{e}_1(\mathsf{nat}_0)$, and $\mathtt{nat}'_0 = (\mathtt{e}_1\mathtt{a} \to \mathtt{a}) \to (\mathtt{e}_1\mathtt{a} \to \mathtt{a})$.

Moreover, if M, N are terms and $n \in \mathbb{N}$, we define $(M)^n N$ by induction on n as follows: $(M)^0 N = N$ and $(M)^{m+1} N = M((M)^m N)$.

We now illustrate our realisability semantics by providing the meaning of the types defined above:

1. $[(\mathtt{a} \sqcap \mathtt{b}) \to \mathtt{a}]_{\beta_1} = \{M \in \mathbb{M}^0 \mid M \twoheadrightarrow^*_\beta \lambda y^0.y^0\}$.
2. It is not possible that $\lambda y^0.y^0 : \langle () \vdash_1 (\mathtt{a} \sqcap \mathtt{b}) \to \mathtt{a} \rangle$.
3. $\lambda y^0.y^0 : \langle () \vdash_2 (\mathtt{a} \sqcap \mathtt{b}) \to \mathtt{a} \rangle$.
4. $[\mathsf{id}_0]_{\beta_3} = \{M \in \mathcal{M}_3^{\oslash} \mid \mathsf{closed}(M) \wedge M \twoheadrightarrow^*_\beta \lambda y^{\oslash}.y^{\oslash}\}$.
5. $[\mathsf{id}_1]_{\beta_3} = \{M \in \mathcal{M}_3^{(1)} \mid \mathsf{closed}(M) \wedge M \twoheadrightarrow^*_\beta \lambda y^{(1)}.y^{(1)}\}$.
6. $[\mathsf{d}]_{\beta_3} = \{M \in \mathcal{M}_3^{\oslash} \mid \mathsf{closed}(M) \wedge M \twoheadrightarrow^*_\beta \lambda y^{\oslash}.y^{\oslash} y^{\oslash}\}$.
7. $[\mathsf{nat}_0]_{\beta_3} = \{M \in \mathcal{M}_3^{\oslash} \mid \mathsf{closed}(M) \wedge (M \twoheadrightarrow^*_\beta \lambda f^{\oslash}.f^{\oslash} \vee (n \geq 1 \wedge M \twoheadrightarrow^*_\beta \lambda f^{\oslash}.\lambda y^{\oslash}.(f^{\oslash})^n y^{\oslash}))\}$.
8. $[\mathsf{nat}_1]_{\beta_3} = \{M \in \mathcal{M}_3^{(1)} \mid \mathsf{closed}(M) \wedge (M \twoheadrightarrow^*_\beta \lambda f^{(1)}.f^{(1)} \vee (n \geq 1 \wedge M \twoheadrightarrow^*_\beta \lambda f^{(1)}.\lambda x^{(1)}.(f^{(1)})^n y^{(1)}))\}$.
9. $[\mathsf{nat}'_0]_{\beta_3} = \{M \in \mathcal{M}_3^{\oslash} \mid \mathsf{closed}(M) \wedge (M \twoheadrightarrow^*_\beta \lambda f^{\oslash}.f^{\oslash} \vee M \twoheadrightarrow^*_\beta \lambda f^{\oslash}.\lambda y^{(1)}.f^{\oslash} y^{(1)})\}$.

3.2. Completeness challenges in $\lambda I^{\mathbb{N}}$

In this document we consider two realisability semantics of types involving E-variables. These semantics are based on a hierarchy of types and terms. Considering how expansions can introduce new substitutions, new expansions and an unbound number of new variables (type variables and E-variables), it was decided to use a hierarchy on types and terms to give meanings to expansions to represent the encapsulation of types by E-variables. An obvious (and naive) approach is to label types and terms with natural numbers. This is the hierarchy we used in $\lambda I^{\mathbb{N}}$. When assigning meanings to types, we ensured that each use of an E-variable in a typing simply changes the indexes of types and terms in the typing and that each E-variable acted as a kind of capsule that isolates parts of the analysed λ-term in a typing. This captured the intuition behind E-variables. However, there are two issues w.r.t. this indexing: it imposes that the type ω should have all possible indexes (which is impossible[5] and hence we eliminated ω from the type systems for $\mathcal{M}_2$) and it implies that the realisability semantics can only be complete when a single E-variable is used (as we will see in this section). In order to understand the challenges of the semantics of E-variables with ω and the idea behind the hierarchy, we first studied two representative intersection type systems for the λI-calculus. The restriction to λI (where in every $(\lambda x.M)$ the variable x must occur free in M) was motivated by not supporting the ω type while preserving the intuitive indexes made of single natural numbers. For $\vdash_1$, the first of these type systems, we showed that subject reduction and hence completeness do not hold.

3.2.1. Completeness for $\vdash_1$ fails

Remark 3.1. (Failure of completeness for $\vdash_1$)
Items 1., 2., and 3. of Example 3.1 show that we can not have a completeness result (a converse of the soundness Lemma 3.3 for closed terms) for $\vdash_1$. To type the term $\lambda y^0.y^0$ by the type $(a \sqcap b) \to a$, we need an elimination rule for $\sqcap$ which we do not have in $\vdash_1$.

Note that failure of completeness for $\vdash_1$ is related to the failure of its subject reduction. So, one might think that since $\vdash_2$, the second type system for $\lambda I^{\mathbb{N}}$, has subject reduction, its semantics is complete. This is not entirely true.

3.2.2. Completeness for $\vdash_2$ fails with more than one E-variable

Remark 3.2. (Failure of completeness for $\vdash_2$ if more than one E-variable are used)
Let a be a type variable, e_1 and e_2 be two distinct expansion variable, and $\mathsf{nat}''_0 = (\mathsf{e}_1\mathsf{a}\to\mathsf{a})\to(\mathsf{e}_2\mathsf{a}\to\mathsf{a})$. Then:

1. $\lambda f^0.f^0 \in [\mathsf{nat}''_0]_{\beta_2}$.
2. it is not possible that $\lambda f^0.f^0 : \langle () \vdash_2 \mathsf{nat}''_0 \rangle$.

Hence $\lambda f^0.f^0 \in [\mathsf{nat}''_0]_{\beta_2}$ but $\lambda f^0.f^0$ is not typable by nat''_0 and we do not have completeness in the presence of more than one expansion variable.

However, we will see that we have completeness for $\vdash_2$ if only one expansion variable is used.

3.2.3. Completeness for $\vdash_2$ with only one E-variable

The problem shown in remark 3.2 comes from the fact that the realisability semantics designed for $\vdash_2$ identifies all expansion variables. In order to give a completeness theorem for $\vdash_2$ we will, in what follows, restrict our system to only one expansion variable. In the rest of this section, we assume that the set ExpVar contains only one expansion variable e_1.

[5] Let us assume that that our type language contains the ω type annotated with integers, i.e., of the form ω^n, then we would need $\mathsf{e}_1\omega^n = \omega^{n+1}$ and $\mathsf{e}_2\omega^n = \omega^{n+1}$, and finally we would have $\mathsf{e}_1\omega^n = \mathsf{e}_2\omega^n$.

The need of one single expansion variable is clear in item 2. of Lemma 3.5 which would fail if we use more than one expansion variable. For example, if $e_1 \neq e_2$ then $(e_1 a)^- = a = (e_2 a)^-$ but $e_1 a \neq e_2 a$. This lemma is crucial for the rest of this section and hence, a single expansion variable is also crucial.

Lemma 3.5. Let $U, V \in \mathsf{ITy}_2$ and $\mathsf{deg}(U) = \mathsf{deg}(V) > 0$.

1. $\mathsf{e}_1 U^- = U$.
2. If $U^- = V^-$ then $U = V$.

Proof:
1. is by induction on U. 2. goes as follows: if $U^- = V^-$ then $\mathsf{e}_1 U^- = \mathsf{e}_1 V^-$ and by 1., $U = V$. $\square$

Despite the difference in the number of considered expansion variables in the completeness proof presented in the current section and the one of Sec. 3.3, both proofs share some similarities. We still write these two proofs independently to illustrate the method and especially since the proof in the current section is far simpler. Furthermore, in the current section we only show the completeness of our semantics w.r.t. β-reduction.

The first step of the proof is to divide $\{y^n \mid y \in \mathsf{Var}_2\}$ into disjoint subset amongst types of order n.

Definition 3.3. Let $U \in \mathsf{ITy}_2$. We define the set of variables DVar_U by induction on $\mathsf{deg}(U)$. If $\mathsf{deg}(U) = 0$ then DVar_U is an infinite set $\{y^0 \mid y \in \mathsf{Var}_2\}$ such that if $U \neq V$ and $\mathsf{deg}(U) = \mathsf{deg}(V) = 0$ then $\mathsf{dj}(\mathsf{DVar}_U, \mathsf{DVar}_V)$. If $\mathsf{deg}(U) = n + 1$ then $\mathsf{DVar}_U = \{y^{n+1} \mid y^n \in \mathsf{DVar}_{U^-}\}$.

Our partition of Var_2 allows useful infinite sets containing type environments that will play a crucial role in one particular type interpretation. These sets and environments are given in the next definition.

Definition 3.4.

- Let $\mathsf{IPreEnv}^n = \{(\!| y^n, U |\!) \mid U \in \mathsf{ITy}_2 \wedge \mathsf{deg}(U) = n \wedge y^n \in \mathsf{DVar}_U\}$ and $\mathsf{BPreEnv}^n = \bigcup_{m \geq n} \mathsf{IPreEnv}^m$ (where "I" stands for "index" and "B" stands for "bound"). Note that $\mathsf{IPreEnv}^n$ and $\mathsf{BPreEnv}^n$ are not type environments because they are not functions.
- If $M \in \mathcal{M}_2$ and $U \in \mathsf{ITy}_2$ then we write $M : \langle \mathsf{BPreEnv}^n \vdash_2 U \rangle$ iff there is a type environment $\Gamma \subseteq \mathsf{BPreEnv}^n$ where $M : \langle \Gamma \vdash_2 U \rangle$.

Now, for every n, we define the set of the good terms of order n which contain some free variable x^i where $x \in \mathsf{Var}_1$ and $i \geq n$.

Definition 3.5. Let $\mathsf{OPEN}^n = \{M \in \mathbb{M}^n \mid x^i \in \mathsf{fv}(M) \wedge x \in \mathsf{Var}_1 \wedge i \geq n\}$.

Obviously, if $x \in \mathsf{Var}_1$ then $\mathsf{VAR}^n_x \subseteq \mathsf{OPEN}^n$.

Here is the crucial β_2-interpretation $\mathbb{I}$ for the proof of completeness:

Definition 3.6. Let $\mathbb{I}$ be the β_2-interpretation defined as follows: for all type variables a, $\mathbb{I}(a) = \mathsf{OPEN}^0 \cup \{M \in \mathcal{M}^0_2 \mid M : \langle \mathsf{BPreEnv}^0 \vdash_2 a \rangle\}$.

The function $\mathbb{I}$ is indeed a β_2-interpretation and the interpretation of a type of order n contains the good terms of order n which are typable in the special environments which are parts of the infinite sets of definition 3.4:

Lemma 3.6. 1. $\mathbb{I}$ is a β_2-interpretation, i.e., for all $a \in \mathsf{TyVar}$, $\mathbb{I}(a)$ is β-saturated and $\forall x \in \mathsf{Var}_1$, $\mathsf{VAR}^0_x \subseteq \mathbb{I}(a) \subseteq \mathbb{M}^0$.

2. If $U \in \mathsf{ITy}_2 \cap \mathsf{GITy}$ and $\mathsf{deg}(U) = n$ then $\mathbb{I}(U) = \mathsf{OPEN}^n \cup \{M \in \mathbb{M}^n \mid M : \langle \mathsf{BPreEnv}^n \vdash_2 U\rangle\}$.

Proof:
We prove 1. by first showing that $\mathbb{I}(a)$ is saturated: if $M \rightarrow\!\!\!\triangleright^*_\beta N$ then if $N \in \mathsf{OPEN}^0$ we prove that $M \in \mathsf{OPEN}^0$ and if $N \in \{M \in \mathcal{M}^0_2 \mid M : \langle \mathsf{BPreEnv}^0 \vdash_2 a\rangle\}$ then $M \in \{M \in \mathcal{M}^0_2 \mid M : \langle \mathsf{BPreEnv}^0 \vdash_2 a\rangle\}$. We then show $\forall x \in \mathsf{Var}_1$. $\mathsf{VAR}^0_x \subseteq \mathbb{I}(a) \subseteq \mathbb{M}^0$. We prove 2. by induction on $U \in \mathsf{GITy}$. □

$\mathbb{I}$ is used to prove completeness (for the proof see the expanded version of this article [22])

Theorem 3.1. (Completeness)
Let $U \in \mathsf{ITy}_2 \cap \mathsf{GITy}$ such that $\mathsf{deg}(U) = n$. The following hold:

1. $[U]_{\beta_2} = \{M \in \mathbb{M}^n \mid M : \langle () \vdash_2 U\rangle\}$.
2. $[U]_{\beta_2}$ is stable by reduction: if $M \in [U]_{\beta_2}$ and $M \rightarrow\!\!\!\triangleright^*_\beta N$ then $N \in [U]_{\beta_2}$.
3. $[U]_{\beta_2}$ is stable by expansion: if $N \in [U]_{\beta_2}$ and $M \rightarrow\!\!\!\triangleright^*_\beta N$ then $M \in [U]_{\beta_2}$.

Proof:
The first item follows by Lemmas 3.6 and 3.3. We obtain the second item using subject reduction and the third one using subject expansion. □

3.3. Completeness for $\lambda^{\mathcal{L}_\mathbb{N}}$

Having understood the challenges of E-variables and the difficulty of representing the type ω using natural numbers as indices for the hierarchy, we moved to the presentation of indices as sequences of natural numbers and we provided our third type system $\vdash_3$. We developed a realisability semantics where we allow the full λ-calculus (i.e., where K-redexes are allowed) indexed with lists of natural numbers, an arbitrary (possibly infinite) number of expansion variables and where ω is present, and we showed its soundness. Now, we show its completeness.

We need the following partition of the set of indexed variables $\{y^L \mid y \in \mathsf{Var}_2\}$.

Definition 3.7. • Let $\mathsf{ITy}^L_3 = \{U \in \mathsf{ITy}_3 \mid \mathsf{deg}(U) = L\}$ and $\mathsf{Var}^L = \{x^L \mid x \in \mathsf{Var}_2\}$.

- We inductively define, for every $U \in \mathsf{ITy}_3$, a set of variables DVar_U as follows:
  - If $\mathsf{deg}(U) = \oslash$ then:
    - DVar_U is an infinite set of indexed variables of degree $\oslash$.
    - If $U \neq V$ and $\mathsf{deg}(U) = \mathsf{deg}(V) = \oslash$ then $\mathsf{dj}(\mathsf{DVar}_U, \mathsf{DVar}_V)$.
    - $\bigcup_{U \in \mathsf{ITy}^\oslash_3} \mathsf{DVar}_U = \mathsf{Var}^\oslash$.
  - If $\mathsf{deg}(U) = i :: L$ then $\mathsf{DVar}_U = \{y^{i::L} \mid y^L \in \mathsf{DVar}_{U-i}\}$.

  Therefore, if $\mathsf{deg}(U) = L$ then $\mathsf{DVar}_U = \{y^L \mid y^\oslash \in \mathsf{DVar}_{U-L}\}$.

Let us now provide some simple results concerning the DVar_U sets:

Lemma 3.7. 1. If $\mathsf{deg}(U) \succeq L$, $\mathsf{deg}(V) \succeq L$, and $U^{-L} = V^{-L}$ then $U = V$.

2. If $\mathsf{deg}(U) = L$ then DVar_U is an infinite subset of Var^L.
3. If $U \neq V$ and $\mathsf{deg}(U) = \mathsf{deg}(V) = L$ then $\mathsf{dj}(\mathsf{DVar}_U, \mathsf{DVar}_V)$.
4. $\bigcup_{U \in \mathsf{ITy}_3^L} \mathsf{DVar}_U = \mathsf{Var}^L$.
5. If $y^L \in \mathsf{DVar}_U$ then $y^{i::L} \in \mathsf{DVar}_{\mathsf{e}_i U}$.
6. If $y^{i::L} \in \mathsf{DVar}_U$ then $y^L \in \mathsf{DVar}_{U^{-i}}$.

Proof:
1. goes as follows: if $L = (n_i)_m$ then we have $U = \mathsf{e}_{n_1} \dots \mathsf{e}_{n_m} U'$ and $V = \mathsf{e}_{n_1} \dots \mathsf{e}_{n_m} V'$; then $U^{-L} = U'$, $V^{-L} = V'$ and $U' = V'$; thus $U = V$. 2., 3. and 4. are by induction on L and using 1. We obtain 5. because $(\mathsf{e}_i U)^{-i} = U$. 6. is by definition. □

The set Var_2 as defined above allows us to give in the next definition useful infinite sets containing type environments that will play a crucial role in one particular type interpretation.

Definition 3.8. • Let $L \in \mathcal{L}_{\mathbb{N}}$. We denote $\mathsf{IPreEnv}^L = \{(\!| y^L, U |\!) \mid U \in \mathsf{ITy}_3^L \wedge y^L \in \mathsf{DVar}_U\}$ and $\mathsf{BPreEnv}^L = \bigcup_{K \succeq L} \mathsf{IPreEnv}^K$. Note that $\mathsf{IPreEnv}^L$ and $\mathsf{BPreEnv}^L$ are not type environments because they are not functions.

- Let $L \in \mathcal{L}_{\mathbb{N}}$, $M \in \mathcal{M}_3$ and $U \in \mathsf{ITy}_3$, we write:
  - $M : \langle \mathsf{BPreEnv}^L \vdash_3 U \rangle$ iff there exists a type environment $\Gamma \subseteq \mathsf{BPreEnv}^L$ such that $M : \langle \Gamma \vdash_3 U \rangle$.
  - $M : \langle \mathsf{BPreEnv}^L \vdash_3^* U \rangle$ iff $M \rightarrow\!\!\!\!\triangleright_{\beta\eta}^* N$ and $N : \langle \mathsf{BPreEnv}^L \vdash_3 U \rangle$.

Let us now provide some results concerning the $\mathsf{BPreEnv}^L$ sets:

Lemma 3.8. 1. If $\Gamma \subseteq \mathsf{BPreEnv}^L$ then $\mathsf{ok}(\Gamma)$.

2. If $\Gamma \subseteq \mathsf{BPreEnv}^L$ then $\mathsf{e}_i \Gamma \subseteq \mathsf{BPreEnv}^{i::L}$.
3. If $\Gamma \subseteq \mathsf{BPreEnv}^{i::L}$ then $\Gamma^{-i} \subseteq \mathsf{BPreEnv}^L$.
4. If $\Gamma_1 \subseteq \mathsf{BPreEnv}^L$, $\Gamma_2 \subseteq \mathsf{BPreEnv}^K$, and $L \preceq K$ then $\Gamma_1 \sqcap \Gamma_2 \subseteq \mathsf{BPreEnv}^L$.

Proof:
1. is by definition. 2. and 3. are by Lemma 3.7. 4. First, by 1., $\Gamma_1 \sqcap \Gamma_2$ is well defined. Also, $\mathsf{BPreEnv}^K \subseteq \mathsf{BPreEnv}^L$. Let $(\Gamma_1 \sqcap \Gamma_2)(x^{L'}) = U_1 \sqcap U_2$ where $\Gamma_1(x^{L'}) = U_1$ and $\Gamma_2(x^{L'}) = U_2$, then $\mathsf{deg}(U_1) = \mathsf{deg}(U_2) = L'$ and $x^{L'} \in \mathsf{DVar}_{U_1} \cap \mathsf{DVar}_{U_2}$. Hence, by Lemma 3.7.3, $U_1 = U_2$ and $\Gamma_1 \sqcap \Gamma_2 = \Gamma_1 \cup \Gamma_2 \subseteq \mathsf{BPreEnv}^L$. □

For every $L \in \mathcal{L}_{\mathbb{N}}$, we define the set of terms of degree L which contain some free variable x^K where $x \in \mathsf{Var}_1$ and $K \succeq L$.

Definition 3.9. For every $L \in \mathcal{L}_{\mathbb{N}}$, let $\mathsf{OPEN}^L = \{M \in \mathcal{M}_3^L \mid x^K \in \mathsf{fv}(M) \wedge x \in \mathsf{Var}_1 \wedge K \succeq L\}$. It is easy to see that, for every $L \in \mathcal{L}_{\mathbb{N}}$ and $x \in \mathsf{Var}_1$, $\mathsf{VAR}_x^L \subseteq \mathsf{OPEN}^L$.

Let us now provide some results on the OPEN^L sets:

Lemma 3.9.

1. $(\mathsf{OPEN}^L)^{+i} = \mathsf{OPEN}^{i::L}$.
2. If $y \in \mathsf{Var}_2$ and $My^K \in \mathsf{OPEN}^L$ then $M \in \mathsf{OPEN}^L$.
3. If $M \in \mathsf{OPEN}^L$, $M \diamond N$, and $L \preceq K = \mathsf{deg}(N)$ then $MN \in \mathsf{OPEN}^L$.
4. If $\mathsf{deg}(M) = L$, $L \preceq K$, $M \diamond N$, and $N \in \mathsf{OPEN}^K$ then $MN \in \mathsf{OPEN}^L$.

Proof:
Easy using Def. 3.9. □

The crucial interpretation $\mathbb{I}$ (the three interpretations $\mathbb{I}_{\beta\eta}$, $\mathbb{I}_\beta$, and $\mathbb{I}_h$ for our three reduction relations) used in the completeness proof is given as follows:

Definition 3.10.

1. Let $\mathbb{I}_{\beta\eta}$ be the $\beta\eta_3$-interpretation defined by: for all type variables a, $\mathbb{I}_{\beta\eta}(a) = \mathsf{OPEN}^\oslash \cup \{M \in \mathcal{M}_3^\oslash \mid M : \langle \mathsf{BPreEnv}^\oslash \vdash_3^* a\rangle\}$.
2. Let $\mathbb{I}_\beta$ be the β_3-interpretation defined by: for all type variables a, $\mathbb{I}_\beta(a) = \mathsf{OPEN}^\oslash \cup \{M \in \mathcal{M}_3^\oslash \mid M : \langle \mathsf{BPreEnv}^\oslash \vdash_3 a\rangle\}$.
3. Let $\mathbb{I}_h$ be the h_3-interpretation defined by: for all type variables a, $\mathbb{I}_h(a) = \mathsf{OPEN}^\oslash \cup \{M \in \mathcal{M}_3^\oslash \mid M : \langle \mathsf{BPreEnv}^\oslash \vdash_3 a\rangle\}$.

The next crucial lemma shows that $\mathbb{I}$ (the three functions $\mathbb{I}_{\beta\eta}$, $\mathbb{I}_\beta$, and $\mathbb{I}_h$) is an interpretation and that the interpretation of a type of order L contains terms of order L which are typable in these special environments which are parts of the infinite sets of Def. 3.8.

Lemma 3.10. Let $r \in \{\beta\eta, \beta, h\}$ and $r' \in \{\beta, h\}$.

1. If $\mathbb{I}_r \in \mathsf{Interp}^{r_3}$ and $a \in \mathsf{TyVar}$ then $\mathbb{I}_r(a) \in \mathsf{SAT}^r$ and $\forall x \in \mathsf{Var}_1.\ \mathsf{VAR}_x^\oslash \subseteq \mathbb{I}_r(a)$.
2. If $U \in \mathsf{ITy}_3$ and $\mathsf{deg}(U) = L$ then $\mathbb{I}_{\beta\eta}(U) = \mathsf{OPEN}^L \cup \{M \in \mathcal{M}_3^L \mid M : \langle \mathsf{BPreEnv}^L \vdash_3^* U\rangle\}$.
3. If $U \in \mathsf{ITy}_3$ and $\mathsf{deg}(U) = L$ then $\mathbb{I}_{r'}(U) = \mathsf{OPEN}^L \cup \{M \in \mathcal{M}_3^L \mid M : \langle \mathsf{BPreEnv}^L \vdash_3 U\rangle\}$.

Proof:
We prove the first item by first showing that $\mathbb{I}_r(a)$ is saturated: if $M \rightarrow\!\!\!\triangleright_r^* N$ then if $N \in \mathsf{OPEN}^\oslash$ we prove that $M \in \mathsf{OPEN}^\oslash$ and if $N \in \{M \in \mathcal{M}_3^\oslash \mid M : \langle \mathsf{BPreEnv}^\oslash \vdash_3^* a\rangle\}$ then $M \in \{M \in \mathcal{M}_3^\oslash \mid M : \langle \mathsf{BPreEnv}^\oslash \vdash_3^* a\rangle\}$. We then show that for all $x \in \mathsf{Var}_1$, $\mathsf{VAR}_x^\oslash \subseteq \mathsf{OPEN}^\oslash \subseteq \mathbb{I}_r(a)$. We prove the second and third items by induction on U. □

Now, we use this crucial $\mathbb{I}$ to establish completeness of our semantics.

Theorem 3.2. (Completeness of $\vdash_3$)
Let $U \in \mathsf{ITy}_3$ such that $\mathsf{deg}(U) = L$.

1. $[U]_{\beta\eta_3} = \{M \in \mathcal{M}_3^L \mid \mathsf{closed}(M) \wedge M \rightarrow\!\!\!\triangleright^*_{\beta\eta} N \wedge N : \langle () \vdash_3 U\rangle\}$.
2. $[U]_{\beta_3} = [U]_{h_3} = \{M \in \mathcal{M}_3^L \mid M : \langle () \vdash_3 U\rangle\}$.
3. $[U]_{\beta\eta_3}$ is stable by reduction: if $M \in [U]_{\beta\eta_3}$ and $M \rightarrow\!\!\!\triangleright_{\beta\eta} N$ then $N \in [U]_{\beta\eta_3}$.

Proof:

1. Let $M \in [U]_{\beta\eta_3}$. Then M is closed and $M \in \mathbb{I}_{\beta\eta}(U)$. By Lemma 3.10.2, $M \in \mathsf{OPEN}^L \cup \{M \in \mathcal{M}_3^L \mid M : \langle \mathsf{BPreEnv}^L \vdash_3^* U\rangle\}$. Since M is closed, $M \notin \mathsf{OPEN}^L$. Hence, $M \in \{M \in \mathcal{M}_3^L \mid M : \langle \mathsf{BPreEnv}^L \vdash_3^* U\rangle\}$ and so, $M \rightarrow\!\!\!\triangleright^*_{\beta\eta} N$ and $N : \langle \Gamma \vdash_3 U\rangle$ where $\Gamma \subseteq \mathsf{BPreEnv}^L$. By Theorem 2.1.2, N is closed and, by Lemma 2.3.2a, $N : \langle () \vdash_3 U\rangle$.

   Conversely, take M closed such that $M \rightarrow\!\!\!\triangleright^*_{\beta} N$ and $N : \langle () \vdash_3 U\rangle$. Let $\mathcal{I} \in \mathsf{Interp}^{\beta_3}$. By Lemma 3.3, $N \in \mathcal{I}(U)$. By Lemma 3.2.1, $\mathcal{I}(U)$ is $\beta\eta$-saturated. Hence, $M \in \mathcal{I}(U)$. Thus $M \in [U]_{\beta\eta_3}$.

2. Let $M \in [U]_{\beta_3}$. Then M is closed and $M \in \mathbb{I}_{\beta}(U)$. By Lemma 3.10.3, $M \in \mathsf{OPEN}^L \cup \{M \in \mathcal{M}_3^L \mid M : \langle \mathsf{BPreEnv}^L \vdash_3 U\rangle\}$. Since M is closed, $M \notin \mathsf{OPEN}^L$. Hence, $M \in \{M \in \mathcal{M}_3^L \mid M : \langle \mathsf{BPreEnv}^L \vdash_3 U\rangle\}$ and so, $M : \langle \Gamma \vdash_3 U\rangle$ where $\Gamma \subseteq \mathsf{BPreEnv}^L$. By Lemma 2.3.2a, $N : \langle () \vdash_3 U\rangle$.

   Conversely, take M such that $M : \langle () \vdash_3 U\rangle$. By Lemma 2.3.2a, M is closed. Let $\mathcal{I} \in \mathsf{Interp}^{\beta_3}$. By Lemma 3.3, $M \in \mathcal{I}(U)$. Thus $M \in [U]_{\beta_3}$.

   It is easy to see that $[U]_{\beta_3} = [U]_{h_3}$.

3. Let $M \in [U]_{\beta\eta_3}$ and $M \rightarrow\!\!\!\triangleright_{\beta\eta} N$. By 1., M is closed, $M \rightarrow\!\!\!\triangleright^*_{\beta\eta} P$, and $P : \langle () \vdash_3 U\rangle$. By confluence Theorem 2.2, there is Q such that $P \rightarrow\!\!\!\triangleright^*_{\beta\eta} Q$ and $N \rightarrow\!\!\!\triangleright^*_{\beta\eta} Q$. By subject reduction Theorem 2.4, $Q : \langle () \vdash_3 U\rangle$. By Theorem 2.1.2, N is closed and, by 1., $N \in [U]_{\beta\eta_3}$.

$\square$

4. Conclusion and future work

Expansion may be viewed to work like a multi-layered simultaneous substitution. Moreover, expansion is a crucial part of a procedure for calculating principal typings and helps support compositional type inference. Because the early definitions of expansion were complicated, expansion variables (E-variables) were introduced to simplify and mechanize expansion. The aim of this document is to give a complete semantics for intersection type systems with expansion variables.

We studied first the $\lambda I^{\mathbb{N}}$-calculus, an indexed version of the λI-calculus. This indexed version was typed using first a basic intersection type system with expansion variables but without an intersection elimination rule, and then using an intersection type system with expansion variables and an elimination rule.

We gave a realisability semantics for both type systems showing that the first type system is not complete in the sense that there are types whose semantics is not the set of $\lambda I^{\mathbb{N}}$-terms having this type. In particular, we showed that $\lambda y^0.y^0$ is in the semantics of $(\mathtt{a} \sqcap \mathtt{b}) \to \mathtt{a}$ but that it is not possible to give $\lambda y^0.y^0$ the type $(\mathtt{a} \sqcap \mathtt{b}) \to \mathtt{a}$ in the type system $\vdash_1$ (see Example 3.1 in Ch. 3.1). The main reason for the failure of completeness in the first system is associated with the failure of the subject reduction property for this first type system. Hence, we moved to the second system which we showed to have the desirable properties of subject reduction and expansion and strong normalisation. However, for this second system, we showed again that completeness fails if we use more than one expansion variable but that completeness succeeds if we restrict the system to a single expansion variable.

In order to overcome the problems of completeness, we changed our realisability semantics from one which uses natural numbers as indices to one that uses lists of natural numbers as indices. The new semantics is more complex and we lose the elegance of the first (especially in being able to define the good terms and good types). However, we consider a third type system for this new indexed calculus and we show that is has all the desirable properties of a type system and it handles all of the λ-calculus (not simply the λI-calculus). We also show that this second semantics is complete when any number (including infinite) of expansion variables is used w.r.t. our third type system. As far as we know, our work constitutes the first study of a realisability semantics of intersection type systems with E-variables and of the difficulties involved.

Note that a restricted version (restricted to normalised types[6]), which we call RCDV, of the well known CDV intersection type system, both systems introduced by Coppo, Dezani and Venneri [7, 8] and recalled by Van Bakel [1], can be embedded in our type system $\vdash_3$ without making use of expansion variables (a more detailed remark can be found in the expanded version of this article [22]). We can then restrict the range of our interpretations (see Def. 3.2) from $\mathcal{M}_3$ to the "space of meaning" $\mathcal{M}_3^{\oslash}$ (see Def. 2.7) which is then the only necessary set because expansion variables are not used and therefore they do not allow one to change the index of terms. Unfortunately, we do not believe that it would be possible to embed RCDV in our system such that we would make use of the expansion variables "as much as possible" (everywhere where an expansion might be needed). For example, if $M : \langle \Gamma \vdash_3 U_1 \sqcap U_2 \rangle$ is derivable from $M : \langle \Gamma \vdash_3 U_1 \rangle$ and $M : \langle \Gamma \vdash_3 U_2 \rangle$ by the intersection introduction rule and we apply the expansion introduction rule to each of the branches of the derivation then we obtain the two following typing judgements: $M^{+i} : \langle \mathsf{e}_i \Gamma \vdash_3 \mathsf{e}_i U \rangle$ and $M^{+j} : \langle \mathsf{e}_j \Gamma \vdash_3 \mathsf{e}_j U \rangle$. If we use two different expansion variables ($i \neq j$) then, given these two new typing judgements, we cannot use the intersection introduction rule because $\mathsf{e}_i U \sqcap \mathsf{e}_j U$ is not a ITy_3 type ($\mathsf{deg}(\mathsf{e}_i U) = i :: \mathsf{deg}(U) \neq j :: \mathsf{deg}(U) = \mathsf{deg}(\mathsf{e}_j U)$). This might be overcome by considering trees instead of lists as indices in our semantics. We let the investigation of such a system to future work.

In the present document we are not interested in a denotational semantics of the presented calculus. We are neither interested in an extensional λ-model interpreting the terms of the untyped λ-calculus. Instead, we are interested in building a realisability semantics by defining sets of realisers (programs satisfying the requirements of some specification) of types. We believe such a model would help high-lighting the relation between terms of the untyped λ-calculus and types involving expansion variables w.r.t. a type system. Moreover, interpreting types in a model helps understanding the meaning of types

[6]Normalised types are types strongly related to normalisable (typable) terms.

(w.r.t. the model) which are defined as purely syntactic forms and are clearly used as meaningful expressions. For example, the integer type (whatever its notation is) is always used as the type of each integer. An arrow type expresses functionality. In that way, models based on λ-models have been built for intersection type systems [16, 3, 10]. In these models, intersection types were interpreted by set-theoretical intersections of meanings. Even though E-variables have been introduced to give a simple formalisation of the expansion mechanism, i.e., as syntactic objects, we are interested in the meaning of such syntactic objects. We are particularly interested in answering a number of questions such as:

1. Can we find a second order function, whose range is the set of λ-terms, and which interprets types involving any kind of expansions (any expansion term and not just expansion variables)?

2. How can we characterise the realisers of a type involving expansion terms?

3. How can the relation between terms and types involving expansion terms be described w.r.t. a type system?

4. How can we extend models such as the one given in Kamareddine and Nour [21] to a type system with expansion?

These questions have not yet been answered. We leave their investigation for future work.

References

[1] Bakel, S. V.: Strict Intersection Types for the Lambda Calculus; a survey, 2011, Located at `http://www.doc.ic.ac.uk/~svb/Research/`.

[2] Barendregt, H. P.: *The Lambda Calculus: Its Syntax and Semantics.*, Revised edition, North-Holland, 1984, ISBN 0-444-88748-1 (hardback).

[3] Barendregt, H. P., Coppo, M., Dezani-Ciancaglini, M.: A Filter Lambda Model and the Completness of Type Assignment., *The Journal of Symbolic Logic*, **48**(4), 1983.

[4] Böhm, C., Ed.: *Lambda-Calculus and Computer Science Theory, Proceedings of the Symposium Held in Rome, March 25-27, 1975*, vol. 37 of *Lecture Notes in Computer Science*, Springer, 1975, ISBN 3-540-07416-3.

[5] Carlier, S., Polakow, J., Wells, J. B., Kfoury, A. J.: System E: Expansion Variables for Flexible Typing with Linear and Non-linear Types and Intersection Types, *ESOP* (D. A. Schmidt, Ed.), 2986, Springer, 2004, ISBN 3-540-21313-9.

[6] Carlier, S., Wells, J. B.: Expansion: the Crucial Mechanism for Type Inference with Intersection Types: A Survey and Explanation, *Electr. Notes Theor. Comput. Sci.*, **136**, 2005, 173–202.

[7] Coppo, M., Dezani-Ciancaglini, M., Venneri, B.: Principal type schemes and λ-calculus semantic., in: *To H. B. Curry: Essays on Combinatory Logic, Lambda Calculus, and Formalism*, J.R. Hindley and J.P. Seldin, 1980, 535–560.

[8] Coppo, M., Dezani-Ciancaglini, M., Venneri, B.: Functional Characters of Solvable Terms, *Mathematische Logik Und Grundlagen der Mathematik*, **27**, 1981, 45–58.

[9] Coquand, T.: Completeness Theorems and lambda-Calculus, *TLCA* (P. Urzyczyn, Ed.), 3461, Springer, 2005, ISBN 3-540-25593-1.

[10] Dezani-Ciancaglini, M., Honsell, F., Alessi, F.: A complete characterization of complete intersection-type preorders, *ACM Trans. Comput. Log.*, **4**(1), 2003, 120–147.

[11] Farkh, S., Nour, K.: Résultats de complétude pour des classes de types du système AF2, *Theoretical Informatics and Applications*, **31**(6), 1998, 513–537.

[12] Gallier, J. H.: On Girard's "Candidats de Reductibilité"., in: *Logic and Computer Science* (P. Odifreddi, Ed.), Academic Press, 1990, 123–203.

[13] Gallier, J. H.: On the correspondence between proofs and λ-terms., *Cahiers du centre de logique*, **8**, 1995, 55–138.

[14] Gallier, J. H.: Proving Properties of Typed λ-Terms Using Realisability, Covers, and Sheaves., *Theoretical Computer Science*, **142**(2), 1995, 299–368.

[15] Gallier, J. H.: Typing Untyped λ-Terms, or Realisability strikes again!., *Annals of Pure and Applied Logic*, **91**, 1998, 231–270.

[16] Hindley, J. R.: The simple semantics for Coppo-Dezani-Sallé types., *Symposium on Programming* (M. Dezani-Ciancaglini, U. Montanari, Eds.), 137, Springer, 1982, ISBN 3-540-11494-7.

[17] Hindley, J. R.: The Completeness Theorem for Typing lambda-Terms., *Theor. Comput. Sci.*, **22**, 1983, 1–17.

[18] Hindley, J. R.: Curry's Types Are Complete with Respect to F-semantics Too, *Theoretical Computer Science*, **22**, 1983, 127–133.

[19] Hindley, J. R.: *Basic Simple Type Theory*, vol. 42 of *Cambridge Tracts in Theoretical Computer Science*, Cambridge University Press, 1997.

[20] Hindley, J. R., Longo, G.: Lambda-Calculus Models and Extensionality., *Zeit. Math. Logik*, **26**, 1980, 289–310.

[21] Kamareddine, F., Nour, K.: A completeness result for a realisability semantics for an intersection type system., *Annals of Pure and Applied Logic*, **146**, 2007, 180–198.

[22] Kamareddine, F., Nour, K., Rahli, V., Wells, J. B.: On Realisability Semantics for Intersection Types and Infinite Expansion Variables, 2008, Located at `http://www.macs.hw.ac.uk/~fairouz/papers/drafts/long-fund-inf-sem.pdf`.

[23] Koletsos, G.: Church-Rosser Theorem for Typed Functional Systems., *Journal of Symbolic Logic*, **50**(3), 1985, 782–790.

[24] Krivine, J.-L.: *Lambda-calcul, types et modèles.*, Masson, 1990.

[25] Labib-Sami, R.: Typer avec (ou sans) types auxilières.

[26] Oosten, J. V.: Realizability: a historical essay, *Mathematical. Structures in Comp. Sci.*, **12**(3), 2002, 239–263, ISSN 0960-1295.

[27] Tait, W. W.: Intensional Interpretations of Functionals of Finite Type I., *The Journal of Symbolic Logic*, **32**(2), 1967, 198–212.