Decidable Properties of Intersection Type Systems

Toshihiko Kurata Masako Takahashi

Department of mathematical and computing sciences
Tokyo institute of technology
kurata@is.titech.ac.jp, masako@is.titech.ac.jp

Abstract

We give positive answers to the following decision problems for the intersection type system $\lambda\wedge$ and its variations; (1) the type checking problem of normalizing terms for $\lambda\wedge$, (2) the inhabitation problem for the system $\lambda\wedge$ without $(\wedge\mathrm{I})$-rule, and (3) the same problem for a typed counterpart of $\lambda\wedge$ (or the intersection type system *à la* Church). Our result (1) contrasts with the well-known negative answer to the type checking problem (of all terms) for $\lambda\wedge$, while (2) and (3) contrast with Urzyczyn's negative answer to the inhabitation problem for $\lambda\wedge$.

1 Introduction

The intersection type system $\lambda\wedge$ (see e.g. [BCD 83], [CC 90]) is an extension of the simple type assignment system $\lambda_{\rightarrow}$, and is introduced to overcome some defects of $\lambda_{\rightarrow}$. Indeed, in the system $\lambda\wedge$ types are invariant under $=_\beta$ and to important terms such as fixed-point combinators meaningful types are assigned. Moreover the notions of strong normalizability, normalizability, and solvability are neatly characterized in the system $\lambda\wedge$.

However as far as decision problems are concerned, what have been known in literature are negative; the type checking problem for $\lambda\wedge$ is undecidable (by Scott's theorem, cf. [Bar 92]), and Urzyczyn's recent result shows that the inhabitation problem for $\lambda\wedge$ is undecidable [Urz 94]. On the other hand, for the simple system $\lambda_{\rightarrow}$ it is well-known that these problems are decidable. Our motivation of the present work is to distinguish rules and other features of $\lambda\wedge$ which make the system undecidable.

First, for $\lambda\wedge$ we prove that the type checking problem of normalizing terms is decidable; that is, the set $\{(\Gamma, A) \mid \Gamma \vdash_{\lambda\wedge} M : A\}$ is decidable when M is normalizing. Next we consider the system $\lambda\wedge$ without $\wedge$-introduction rule, and prove that the inhabitation problem for it is decidable. We also study a variation of the system $\lambda\wedge$ which might be considered as a typed counterpart of $\lambda\wedge$ (or the intersection type system *à la* Church). For the system, among others we prove that the inhabitation problem is decidable.

The paper is organized as follows. In section 2, we prove some fundamental properties of intersection types. In particular, we show that any two types have supremum as well as infimum in the set of intersection types (with respect to the subtype relation $\leq$).

In section 3, we define a typed counterpart $\lambda\wedge'$ of $\lambda\wedge$, and prove relationship between the systems $\lambda\wedge'$, $\lambda\wedge'$ without $\wedge$-introduction rule, and $\lambda\wedge$ without $\wedge$-introduction rule. Also an approximation theorem for $\lambda\wedge'$ is proved.

In section 4, we prove characterization theorems of judgements, one for $\lambda\wedge$ and the other for $\lambda\wedge'$. Then from the theorems we obtain positive answers to type checking problems; the problem of normalizing terms for the system $\lambda\wedge$, and that of β-nf's for the system $\lambda\wedge'$.

In section 5, we study inhabitation problems. From the results in section 3, we see that our inhabitation problems mentioned above can be reduced to the emptiness problem of the set $\{M \text{ in } \beta\text{-nf} \mid \Gamma \vdash_{\lambda\wedge'} M : A\}$. Then based on the characterization theorem for $\lambda\wedge'$ in section 4 the latter problem is shown to be reducible to the emptiness problem of context-free grammars, which is well-known to be decidable.

Some proofs of theorems in sections 3 and 5 are given in Appendix.

2 Intersection types

The set $T_\wedge$ of *types* in intersection type systems consists of atomic types including ω, arrow types $(A \to B)$, and intersection types $(A \wedge B)$ where A, B are types. As usual we will omit some parentheses; for example, we write $A_1 \to A_2 \to \cdots \to A_n$ for $(A_1 \to (A_2 \to \cdots \to (A_{n-1} \to A_n)\cdots))$. *Subtype relation* $\leq$ between types is defined by the following axioms and rules:

$$\begin{array}{ll}
A \leq A \wedge A, & (A \to B) \wedge (A \to C) \leq A \to B \wedge C, \\
A_1 \wedge A_2 \leq A_i \ (i = 1,2), & A \leq B, B \leq C \Longrightarrow A \leq C, \\
A \leq \omega, & A \leq B, C \leq D \Longrightarrow A \wedge C \leq B \wedge D, \\
\omega \leq \omega \to \omega, & A \leq B, C \leq D \Longrightarrow B \to C \leq A \to D.
\end{array}$$

Note that $\leq$ is a pre-order satisfying $(A_1 \to B_1) \wedge (A_2 \to B_2) \leq A_1 \wedge A_2 \to B_1 \wedge B_2$, and the equivalence relation $=$ induced by $\leq$ (i.e., $A = B \Longleftrightarrow_{def} A \leq B$ and $B \leq A$) satisfies: $A = A \wedge A$, $A \wedge (B \wedge C) = (A \wedge B) \wedge C$, $A \wedge B = B \wedge A$, $(A \to B) \wedge (A \to C) = A \to B \wedge C$, and $A \to B = \omega$ if and only if $B = \omega$. Note also that $A_1 \wedge A_2$ is an infimum of A_1 and A_2. In this paper, for the sake of simplicity, we identify (in notation $\equiv$) conjunctions of types $A_1, A_2, \ldots, A_m$ in any order and any association, and write it as $\bigwedge_{i=1}^m A_i$. We also use notations $\bigwedge_{i \in I} A_i$ or $\bigwedge\{A_i \mid i \in I\}$ for any finite set I. (In particular, when $I = \emptyset$, they stand for ω.) We use $A, B, \ldots$ as metavariables ranging over types, $a, b, \ldots$ for atomic types different from ω, and $I, J, \ldots$ for finite sets (of indices).

The set NT (of *normal types*) and its subset $\mathsf{NT}^{\to}$ (of *arrow normal types*) are defined by simultaneous recursion, as follows:

1. If $A_1, A_2, \ldots, A_n \in \mathsf{NT}$ $(n \geq 0)$, then $A_1 \to \cdots \to A_n \to a \in \mathsf{NT}^{\to}$.
2. If $A_1, A_2, \ldots, A_n \in \mathsf{NT}^{\to}$ $(n \geq 0)$, then $\bigwedge_{i=1}^n A_i \in \mathsf{NT}$. (In particular, $\omega \in \mathsf{NT}$.)

Normal types satisfy the following properties (cf. [Hin 82]).

Lemma 2.1 If $A_i, B_j \in \mathsf{NT}^{\to}$ $(i \in I, j \in J)$, then

$$\bigwedge_{i\in I} A_i \le \bigwedge_{j\in J} B_j \Longleftrightarrow \forall j\, \exists i\ (A_i \le B_j). \qquad \square$$

Lemma 2.2 If $A, B \in \mathsf{NT}^{\rightarrow}$, then

$$A \le B \Longleftrightarrow \text{ for some } p \ge 0,\ a,\ A_i\text{'s and } B_i\text{'s,} \quad \begin{cases} A \equiv A_1 \to \cdots \to A_p \to a, & B \equiv B_1 \to \cdots \to B_p \to a \\ \text{and } B_i \le A_i\ (i = 1, ..., p). \end{cases} \qquad \square$$

In next definition and lemma, we introduce the notion of normal type A^* of A, which is shown to be the unique representative of the equivalence class containing A.

Definition 2.3 A *normal type* A^* of A is defined recursively, as follows:

1. $a^* \equiv a$.
2. $(\bigwedge_{i\in I} A_i)^* \equiv \bigwedge_{i\in J} A_i^*$ where J is a minimum subset of I such that $\bigwedge_{i\in J} A_i^* = \bigwedge_{i\in I} A_i^*$.
3. $(A \to B)^* \equiv \begin{cases} \bigwedge_{i\in I}(A^* \to B_i) & \text{if } B^* \equiv \bigwedge_{i\in I} B_i \not\equiv \omega, B_i \in \mathsf{NT}^{\rightarrow}\ (i \in I), \\ \omega & \text{if } B^* \equiv \omega. \end{cases}$ $\square$

Note that the condition $\bigwedge_{j\in J} B_j = \bigwedge_{j\in J'} B_j$ is decidable by Lemmas 2.1 and 2.2, which enables us to construct A^* from A effectively.

Lemma 2.4 (1) $A^* \in \mathsf{NT}$; and $A = A^*$.
(2) If $A^* \equiv \bigwedge_{i\in I} A_i$ and $A_i \in \mathsf{NT}^{\rightarrow}$ $(i \in I)$, then $\forall i, i' \in I\ (A_i \le A_{i'} \Longrightarrow i = i')$.
(3) $A = B \Longleftrightarrow A^* \equiv B^*$.
Proof. (1) and (2) are by induction on types. (3) $A = B$ if and only if $A^* = B^*$ by (1). So it suffices to show that $A^* = B^*$ implies $A^* \equiv B^*$. We will prove it by induction on the number of occurrences of atomic types in A^* and B^*.

Suppose $A^* = B^*$. If $A^*, B^* \in \mathsf{NT}^{\rightarrow}$, then they are of the form $A^* \equiv A_1^* \to \cdots \to A_n^* \to a, B^* \equiv B_1^* \to \cdots \to B_n^* \to a$ where $A_i^* = B_i^*$ $(i = 1, ..., m)$ (cf. Lemma 2.2), which implies $A_i^* \equiv B_i^*$ $(i = 1, ..., m)$ by induction hypothesis; hence $A^* \equiv B^*$. For the general case, assume $A^* \equiv \bigwedge_{i\in I} A_i^*$ and $B^* \equiv \bigwedge_{j\in J} B_j^*$ where $A_i^*, B_j^* \in \mathsf{NT}^{\rightarrow}$ $(i \in I, j \in J)$. Then by Lemma 2.1 $\forall i \in I\ \exists j \in J\ \exists i' \in I\ (A_{i'}^* \le B_j^* \le A_i^*)$; hence by (2) we get $i' = i$ and so $A_i^* = B_j^*$. This implies $A_i^* \equiv B_j^*$ by induction hypothesis. Thus we get $\{A_i^* \mid i \in I\} \subseteq \{B_j^* \mid j \in J\}$. Likewise for the converse; hence $A^* \equiv \bigwedge_{i\in I} A_i^* \equiv \bigwedge_{j\in J} B_j^* \equiv B^*$. $\square$

Corollary 2.5 The binary relation $A \le B$ is decidable.
Proof. Check whether $A^* \le B^*$ or not, by using Lemmas 2.1 and 2.2. $\square$

The properties of normal types described in Lemmas 2.1 and 2.2 can be generalized to types, as follows.

Lemma 2.6 Suppose $\bigwedge_{i\in I}(A_i \to A_i') \le B$. Then
(1) $B^* \equiv \bigwedge_{j\in J}(C_j \to C_j')$ for some J, C_j's and C_j''s. In particular, if $B \in \mathsf{NT}^{\rightarrow}$ then $B \equiv C \to C'$ for some C and C'.
(2) If $B \equiv C \to C' \in \mathsf{NT}^{\rightarrow}$, then $\exists i \in I\ (C \le A_i$ and $A_i' \le C')$.
(3) If $B \equiv C \to C' \ne \omega$, then $\exists I'\ (\subseteq I, \ne \emptyset)\ (C \le \bigwedge_{i\in I'} A_i$ and $\bigwedge_{i\in I'} A_i' \le C')$. In particular, if $A \to A' \le C \to C' \ne \omega$, then $C \le A$ and $A' \le C'$.

Proof. (1) follows immediately from the special case with $B \in \mathsf{NT}^{\rightarrow}$. To prove the special case, suppose $A_i'^* \equiv \bigwedge_{k\in K_i} A'_{i,k}$ with $A'_{i,k} \in \mathsf{NT}^{\rightarrow}$ $(i \in I, k \in K_i)$. Then since $\bigwedge_{i\in I}\bigwedge_{k\in K_i}(A_i^* \rightarrow A'_{i,k}) = \bigwedge_{i\in I}(A_i \rightarrow A_i') \leq B$ and $B \in \mathsf{NT}^{\rightarrow}$, we know from Lemmas 2.1 and 2.2 that B is of the form $B \equiv C \rightarrow C'$, and moreover $C \leq A_i$ and $A_i' \leq A'_{i,k} \leq C'$ for some $i \in I$ and $k \in K_i$. This proves (1) and (2).

To see (3), suppose $C'^* \equiv \bigwedge_{j\in J} C_j'$ where $C_j' \in \mathsf{NT}^{\rightarrow}$ $(j \in J)$. Then our assumption implies $\bigwedge_{i\in I}(A_i \rightarrow A_i') \leq \bigwedge_{j\in J}(C^* \rightarrow C_j') \leq C^* \rightarrow C_j'$ for each j. Then by applying (2) we get $\forall j \in J, \exists i_j \in I\ (C^* \leq A_{i_j}$ and $A'_{i_j} \leq C_j')$. Let $I' = \{i_j | j \in J\}$. Then $I' \neq \emptyset$ since $J \neq \emptyset$, and it satisfies $C \leq \bigwedge_{j\in J} A_{i_j} \equiv \bigwedge_{i\in I'} A_i$ and $\bigwedge_{i\in I'} A_i' \equiv \bigwedge_{j\in J} A'_{i_j} \leq \bigwedge_{j\in J} C_j' \leq C'$. □

Lemma 2.7 Suppose $A \leq C \rightarrow C' \neq \omega$. Then $A^* \equiv \bigwedge_{i\in I}(A_i \rightarrow A_i') \wedge A'$, $C \leq \bigwedge_{i\in I} A_i$ and $\bigwedge_{i\in I} A_i' \leq C'$ for some $I \neq \emptyset$, A_i's, A_i''s and A'.

Proof. Let $A^* \equiv \bigwedge_{i\in I} B_i$ and $C'^* \equiv \bigwedge_{j\in J} C_j'$ where $B_i, C_j' \in \mathsf{NT}^{\rightarrow}$ $(i \in I, j \in J)$. Then by assumption $\bigwedge_{i\in I} B_i \leq (C \rightarrow C')^* \equiv \bigwedge_{j\in J}(C^* \rightarrow C_j')$; hence by Lemma 2.1 $\forall j \in J, \exists i_j \in I\ (B_{i_j} \leq C^* \rightarrow C_j')$. Since $B_{i_j} \in \mathsf{NT}^{\rightarrow}$, this together with Lemma 2.2 implies $B_{i_j} \equiv A_{i_j} \rightarrow A'_{i_j}, C^* \leq A_{i_j}$ and $A'_{i_j} \leq C_j'$ $(j \in J)$ for some A_{i_j} and A'_{i_j}. Then as before for $I' = \{i_j | j \in J\}$ $(\neq \emptyset)$, we have $C \leq \bigwedge_{i\in I'} A_i$ and $\bigwedge_{i\in I'} A_i' \leq \bigwedge_{j\in J} C_j' \leq C'$, and moreover $A^* \equiv \bigwedge_{i\in I} B_i \equiv \bigwedge_{i\in I'}(A_i \rightarrow A_i') \wedge A'$ where $A' \equiv \bigwedge_{i\in I-I'} B_i$. □

We can extend these results to types with two or more consecutive arrows. For example, we have the following generalization of Lemma 2.7 and its converse.

Corollary 2.8 Suppose $C \neq \omega$. Then

$$A \leq C_1 \rightarrow \cdots \rightarrow C_q \rightarrow C \iff \begin{array}{l} \text{for some } I \neq \emptyset,\ A_{i,j}\text{'s},\ A_i\text{'s and } A', \\ \left\{ \begin{array}{l} A^* \equiv \bigwedge_{i\in I}(A_{i,1} \rightarrow \cdots \rightarrow A_{i,q} \rightarrow A_i) \wedge A', \\ C_j \leq \bigwedge_{i\in I} A_{i,j} \ (j = 1, \ldots, q), \text{ and } \bigwedge_{i\in I} A_i \leq C. \end{array} \right. \end{array}$$

Proof. (Only if part) By induction on q, using Lemma 2.7. (If part) $A = A^* \leq \bigwedge_{i\in I} A_{i,1} \rightarrow \cdots \rightarrow \bigwedge_{i\in I} A_{i,q} \rightarrow \bigwedge_{i\in I} A_i \leq C_1 \rightarrow \cdots \rightarrow C_q \rightarrow C$. □

In $\mathrm{T}_\wedge$, types have infima by definition. We will show that they also have suprema (with respect to the subtype relation $\leq$). For $A, B \in \mathsf{NT}^{\rightarrow}$, define

$$A \vee B \equiv \begin{cases} A_1 \wedge B_1 \rightarrow \cdots \rightarrow A_n \wedge B_n \rightarrow a & \\ \quad \text{if } A \equiv A_1 \rightarrow \cdots \rightarrow A_n \rightarrow a, B \equiv B_1 \rightarrow \cdots \rightarrow B_n \rightarrow a & \\ \quad \text{for some } n \geq 0,\ A_i\text{'s},\ B_i\text{'s and } a, & \\ \omega \quad \text{otherewise.} & \end{cases}$$

We extend the definition to arbitrary $A, B \in \mathrm{T}_\wedge$ by $A \vee B \equiv \bigwedge\{A_i \vee B_j \mid i \in I, j \in J\}$ where $A^* \equiv \bigwedge_{i\in I} A_i$ and $B^* \equiv \bigwedge_{j\in J} B_j$ with $A_i, B_j \in \mathsf{NT}^{\rightarrow}$ $(i \in I, j \in J)$.

Theorem 2.9 For any $A, B \in \mathrm{T}_\wedge$, $A \vee B$ is a supremum of A and B; i.e., $\forall C \in \mathrm{T}_\wedge\ (A \leq C$ and $B \leq C \iff A \vee B \leq C)$.

Proof. (If part) is clear from the definition. (Only if part) When $A, B \in \mathsf{NT}^{\rightarrow}$, it is immediate from Lemmas 2.1 and 2.2. In other cases, suppose $A^* \equiv \bigwedge_{i\in I} A_i$,

$B^* \equiv \bigwedge_{j\in J} B_j$ and $C^* \equiv \bigwedge_{k\in K} C_k$ with $A_i, B_j, C_k \in \mathsf{NT}^{\rightarrow}$ $(i \in I, j \in J, k \in K)$. Then by using Lemma 2.1 and the case above for $\mathsf{NT}^{\rightarrow}$ we have

$$\begin{aligned} A \leq C \text{ and } B \leq C &\implies \forall k\,(\exists i\,(A_i \leq C_k) \text{ and } \exists j\,(B_j \leq C_k)) \\ &\implies \forall k\, \exists i\, \exists j\,(A_i \vee B_j \leq C_k) \\ &\implies \bigwedge\{A_i \vee B_j \mid i \in I, j \in J\} \leq \bigwedge_{k\in K} C_k \\ &\implies A \vee B \leq C. \qquad \square \end{aligned}$$

We can easily see that distributive laws and the equality $(A \to B) \vee (C \to D) = A \wedge C \to B \vee D$ hold.

3 Intersection type systems $\lambda\wedge$ and $\lambda\wedge'$

In the standard intersection type system $\lambda\wedge$ (cf. [BCD 83], [CC 90], [Bar 92]), *type assignments* or *judgements* $\Gamma \vdash M : A$ are derived from *axioms* (var) and (ω) by applying *inference rules* ($\to$ I), ($\to$ E), ($\wedge$I) and ($\leq$) below. Here the *statement* $M : A$ consists of the *subject* M which is a (type-free) λ-term and the *predicate* A which is a type (in $\mathrm{T}_\wedge$). The *basis* Γ is a finite set of statements whose subjects are term variables distinct each other.

$$\text{(var)}\quad \Gamma \vdash x : A \quad (x : A \in \Gamma) \qquad\qquad (\omega)\quad \Gamma \vdash M : \omega$$

$$(\to \mathrm{I})\quad \frac{\Gamma, x : B \vdash M : A}{\Gamma' \vdash \lambda x.M : B \to A} \qquad\qquad (\to \mathrm{E})\quad \frac{\Gamma \vdash M : A \to B \quad \Gamma \vdash N : A}{\Gamma \vdash MN : B}$$

$$(\wedge \mathrm{I})\quad \frac{\Gamma \vdash M : A \quad \Gamma \vdash M : B}{\Gamma \vdash M : A \wedge B} \qquad\qquad (\leq)\quad \frac{\Gamma \vdash M : A \quad A \leq B}{\Gamma \vdash M : B}$$

In ($\to$ I)-rule, Γ' is any basis including Γ (so that 'weakening' is derived).

In this paper we consider the system $\lambda\wedge$ and its variations including what one might call a typed counterpart of $\lambda\wedge$, or the intersection type system *à la* Church. In the latter system, which we denote by $\lambda\wedge'$, terms are not type-free but typed; that is, those in which types are assigned to bound variables. Judgements in $\lambda\wedge'$ are derived by the same axioms and rules as above, except that ($\to$ I) is now replaced by the following.

$$(\to \mathrm{I})'\quad \frac{\Gamma, x : B \vdash M : A}{\Gamma' \vdash (\lambda x : B.M) : B \to A}$$

We note that the system $\lambda\wedge'$ is a subsystem of the programming language 'Forsythe' introduced by Reynolds [Rey 88]. In next lemma we show that ($\wedge$I)'-rule is redundant in the system $\lambda\wedge'$, and hence $\lambda\wedge'$ is comparable with the system $\lambda\wedge$ without ($\wedge$I)-rule.[1]

Lemma 3.1 (1) $\Gamma \vdash_{\lambda\wedge'} M : A \Longleftrightarrow \Gamma \vdash_{\lambda\wedge' - (\wedge\mathrm{I})'} M : A$.
(2) $\exists N(\Gamma \vdash_{\lambda\wedge'-(\wedge\mathrm{I})'} N : A$ and $|N| \equiv M)$[2] $\Longleftrightarrow \Gamma \vdash_{\lambda\wedge-(\wedge\mathrm{I})} M : A$.
Proof. By induction on terms ($\Longrightarrow$ of (1)) and on derivations. $\square$

[1] We write $\lambda\wedge$-($\wedge$I) for the system $\lambda\wedge$ without ($\wedge$I)-rule, and similarly for $\lambda\wedge'$-($\wedge$I)'.
[2] By $|N|$ we mean the type erasure of a typed term N.

The lemma shows that the system $\lambda\wedge'$ is considerably weaker than the system $\lambda\wedge$, though it is much stronger than the simple type system; for example, $\lambda x : A.xx$ is not typable in $\lambda_{\rightarrow}$, but $\vdash_{\lambda\wedge'} \lambda x : A.xx : A \rightarrow B$ when $A \equiv B \wedge (B \rightarrow B)$. More generally, we can prove that for any type-free λ-term M in β-nf, there exists a typed term N such that $M \equiv |N|$ and $\Gamma \vdash_{\lambda\wedge'} N : A$ for some Γ and ω-free type A. It can also be verified that β-reduction and η-reduction preserve types in $\lambda\wedge'$, although neither β-conversion nor η-conversion does. Indeed, for $\Gamma = \{x : a \wedge b\}$ and $M \equiv (\lambda y : a.y)x \rightarrow_\beta x$ we have $\Gamma \vdash_{\lambda\wedge'} M : a$, but $\Gamma \not\vdash_{\lambda\wedge'} M : a \wedge b$, while for $\Delta = \{x : a \rightarrow a\}$ and $N \equiv \lambda y : a \wedge b.xy \rightarrow_\eta x$ we have $\Delta \vdash_{\lambda\wedge'} N : a \wedge b \rightarrow a$ but $\Delta \not\vdash_{\lambda\wedge'} N : a \rightarrow a$.

When we define the notion of finite approximation ($\sqsubseteq$ in notation) as usual (see e.g., [DM 86]), we can prove the following (one-sided) approximation theorem for the system $\lambda\wedge'$.

Theorem 3.2 $\Gamma \vdash_{\lambda\wedge'} M : A \Longrightarrow \exists P \sqsubseteq M\ (\Gamma \vdash_{\lambda\wedge'} P : A)$.
Proof. See Appendix A. □

The converse however does not hold (cf. the counterexample above for β-conversion).

4 Type checking problems for $\lambda\wedge$ and $\lambda\wedge'$

In this section, for each of the systems $\lambda\wedge$ and $\lambda\wedge'$ we provc a characterization theorem of judgements $\Gamma \vdash M : A$ where M is in β-nf, and based on the theorem we give positive answers to the type checking problem of normalizing terms for $\lambda\wedge$ and that of β-nf's for $\lambda\wedge'$.

Lemma 4.1 (Generation lemma for $\lambda\wedge$)
(1) If $A \not\equiv \omega$, then $\Gamma \vdash_{\lambda\wedge} x : A \Longleftrightarrow \exists B\ (x : B \in \Gamma$ and $B \leq A)$.
(2) $\Gamma \vdash_{\lambda\wedge} MN : A \Longleftrightarrow \exists B\ (\Gamma \vdash_{\lambda\wedge} M : B \rightarrow A$ and $\Gamma \vdash_{\lambda\wedge} N : B)$.
(3) If $A \not\equiv \omega$, then $\Gamma \vdash_{\lambda\wedge} \lambda x.M : A \Longleftrightarrow$ for some m, A_i's and A_i''s $[A^* \equiv \bigwedge_{i=1}^{m}(A_i \rightarrow A_i')$ and $(\Gamma \setminus x), x : A_i \vdash_{\lambda\wedge} M : A_i'$ $(i = 1, ..., m)$].[3] □

Theorem 4.2 If $A \not\equiv \omega$, then

$$
\begin{array}{l}
\Gamma \vdash_{\lambda\wedge} \lambda x_1 \cdots x_p.xM_1M_2\cdots M_q : A \Longleftrightarrow \\
\quad \text{for some } m, A_{i,j}\text{'s and } A_i\text{'s,} \\
\quad \left\{\begin{array}{l}
A^* \equiv \bigwedge_{i=1}^{m}(A_{i,1} \rightarrow \cdots \rightarrow A_{i,p} \rightarrow A_i), \\
\text{for each } i = 1, ..., m, \\
\text{for some } C,\ n \geq 1,\ C_{k,j}\text{'s, } C_k\text{'s, } C'\text{, and } \Gamma' \\
\left\{\begin{array}{l}
x : C \in \Gamma',\ \Gamma' = (\Gamma \setminus x_1, ..., x_p) \cup \{x_1 : A_{i,1}, ..., x_p : A_{i,p}\}, \\
C^* \equiv \bigwedge_{k=1}^{n}(C_{k,1} \rightarrow \cdots \rightarrow C_{k,q} \rightarrow C_k) \wedge C', \\
\Gamma' \vdash_{\lambda\wedge} M_j : \bigwedge_{k=1}^{n} C_{k,j}\ \ (j = 1, ..., q), \bigwedge_{k=1}^{n} C_k \leq A_i.
\end{array}\right.
\end{array}\right.
\end{array}
$$

Proof. From Lemma 4.1 (1), (2) and Corollary 2.8, we have

$$
\begin{array}{l}
\Gamma \vdash_{\lambda\wedge} xM_1M_2\cdots M_q : A \Longleftrightarrow \\
\quad \text{for some } C,\ n \geq 1,\ C_{i,j}\text{'s, } C_i\text{'s and } C',
\end{array}
$$

[3] By $\Gamma \setminus x_1, ..., x_n$ we mean the basis consisting of statements in Γ whose subjects are different from $x_1, ..., x_n$.

$$\left\{\begin{array}{l} x : C \in \Gamma,\ C^* \equiv \bigwedge_{i=1}^{n}(C_{i,1} \to \cdots \to C_{i,q} \to C_i) \wedge C', \\ \Gamma \vdash_{\lambda\wedge} M_j : \bigwedge_{i=1}^{n} C_{i,j}\ \ (j = 1, ..., q), \bigwedge_{i=1}^{n} C_i \leq A. \end{array}\right.$$

Then by applying Lemma 4.1 (3) we get the theorem. □

Theorem 4.3 For a basis Γ, a term M in β-nf and a type A, whether $\Gamma \vdash_{\lambda\wedge} M : A$ or not is decidable.

Proof. The proof is by induction on M. First, if $A = \omega$, then the judgement $\Gamma \vdash_{\lambda\wedge} M : A$ always holds. Otherwise, suppose $M \equiv \lambda x_1 \cdots x_p.xM_1M_2\cdots M_q$. Then by Theorem 4.2, $\Gamma \vdash_{\lambda\wedge} M : A$ holds true if and only if there exist m (≥ 1), $A_{i,j}$'s, A_i's, C_i's, n_i's (≥ 1), $C_{i,k,j}$'s, $C_{i,k}$'s and C_i''s such that

1. $A^* \equiv \bigwedge_{i=1}^{m}(A_{i,1} \to \cdots \to A_{i,p} \to A_i)$,
2. $(x : C_i) \in \Gamma_i'$ $(i = 1, ..., m)$,
3. $C_i^* \equiv \bigwedge_{k=1}^{n_i}(C_{i,k,1} \to \cdots \to C_{i,k,q} \to C_{i,k}) \wedge C_i'$ $(i = 1, ..., m)$,
4. $\Gamma_i' \vdash_{\lambda\wedge} M_j : \bigwedge_{k=1}^{n_i} C_{i,k,j}$ $(i = 1, ..., m,\ j = 1, ..., q)$,
5. $\bigwedge_{k=1}^{n_i} C_{i,k} \leq A_i$ $(i = 1, ..., m)$

where $\Gamma_i' = (\Gamma \setminus x_1, ..., x_p) \cup \{x_1 : A_{i,1}, ..., x_p : A_{i,p}\}$. Note that there are only finitely many combinations of $m, A_{i,j}$'s, A_i's, C_i's, n_i's, $C_{i,k,j}$'s, $C_{i,k}$'s, C_i''s and Γ_i''s satisfying conditions 1, 2 and 3, and moreover that they can be obtained effectively from Γ, M, and A. For each of these combinations it is possible to check whether it satisfies the conditions 4 and 5 by induction hypothesis and Corollary 2.5. □

Corollary 4.4 When M is normalizing, $\Gamma \vdash_{\lambda\wedge} M : A$ or not is decidable.

Proof. Immediate from Theorem 4.3 because β-nf's of normalizing terms can effectively be obtained, and types are invariant under $=_\beta$ in $\lambda\wedge$. □

Similar discussion as lemma 4.1 through theorem 4.3 can be carried out to the system $\lambda\wedge'$, yielding the following.

Lemma 4.5 (Generation lemma for $\lambda\wedge'$)
(1) If $A \neq \omega$, then $\Gamma \vdash_{\lambda\wedge'} x : A \iff \exists B\ (x : B \in \Gamma$ and $B \leq A)$.
(2) $\Gamma \vdash_{\lambda\wedge'} MN : A \iff \exists B\ (\Gamma \vdash_{\lambda\wedge'} M : B \to A$ and $\Gamma \vdash_{\lambda\wedge'} N : B)$.
(3) If $A \neq \omega$, then $\Gamma \vdash_{\lambda\wedge'} (\lambda x : B.M) : A \iff$ for some m, A_i's and A_i''s [$A^* \equiv \bigwedge_{i=1}^{m}(A_i \to A_i')$, $(\Gamma \setminus x), x : B \vdash_{\lambda\wedge'} M : \bigwedge_{i=1}^{m} A_i'$, and $\bigvee_{i=1}^{m} A_i \leq B$.] □

Theorem 4.6 If $A \neq \omega$, then

$$\Gamma \vdash_{\lambda\wedge'} (\lambda x_1 : B_1 \cdots x_p : B_p.xM_1M_2\cdots M_q) : A \iff$$
for some $m, A_{i,j}$'s, A_i's, C, $n \geq 1$, $C_{i,j}$'s, C_i's, C' and Γ'

$$\left\{\begin{array}{l} A^* \equiv \bigwedge_{i=1}^{m}(A_{i,1} \to \cdots \to A_{i,p} \to A_i), \\ (x : C) \in \Gamma',\ \Gamma' = (\Gamma \setminus x_1, ..., x_p) \cup \{x_1 : B_1, ..., x_p : B_p\}, \\ C^* \equiv \bigwedge_{i=1}^{n}(C_{i,1} \to \cdots \to C_{i,q} \to C_i) \wedge C', \\ \Gamma' \vdash_{\lambda\wedge'} M_j : \bigwedge_{i=1}^{n} C_{i,j}\ \ (j = 1, ..., q), \\ \bigvee_{i=1}^{m} A_{i,k} \leq B_k\ \ (k = 1, ..., p),\ \ \bigwedge_{i=1}^{n} C_i \leq \bigwedge_{i=1}^{m} A_i. \end{array}\right.$$ □

Theorem 4.7 When M is in β-nf, $\Gamma \vdash_{\lambda\wedge'} M : A$ or not is decidable. □

5 Inhabitation problems for $\lambda\wedge-(\wedge\mathrm{I})$ and $\lambda\wedge'$

In this section, we prove that the inhabitation problems for $\lambda\wedge-(\wedge\mathrm{I})$ and $\lambda\wedge'$ are decidable. First we note that the problems can be reduced to the emptiness problem of the set $\{M \text{ in } \beta\text{-nf} \mid \Gamma \vdash_{\lambda\wedge'} M : A\}$.

Theorem 5.1 The followings are equivalent:

1. $\exists M(\Gamma \vdash_{\lambda\wedge-(\wedge\mathrm{I})} M : A)$.
2. $\exists M(\Gamma \vdash_{\lambda\wedge'} M : A)$.
3. $\exists M$ in β-nf $(\Gamma \vdash_{\lambda\wedge'} M : A)$.

Proof. $1 \Longleftrightarrow 2$ is immediate from Lemma 3.1. $2 \Longrightarrow 3$ is from Theorem 3.2 and the fact that $\Gamma \vdash_{\lambda\wedge'} P : A$ with P in $\beta\bot$-nf implies $\Gamma \vdash_{\lambda\wedge'} N : A$ for the λ-term N in β-nf which is obtained from P by replacing $\bot$ with a variable. $3 \Longrightarrow 2$ is obvious. □

Next we observe that the emptiness of $\{M \text{ in } \beta\text{-nf} \mid \Gamma \vdash_{\lambda\wedge'} M : A\}$ can be reduced to the same problem for the following subsystem $\lambda\wedge''$ of $\lambda\wedge'$: The set of term variables in $\lambda\wedge''$ is restricted to the set $\{x_A \mid A \in \mathrm{T}_\wedge\}$ which is in one-to-one correspondence with the set $\mathrm{T}_\wedge$ of types, and each type A is assigned to x_A but no other variables. In other words, bases of the system $\lambda\wedge''$ are finite subsets of $\{x_A : A \mid A \in \mathrm{T}_\wedge\}$, and abstraction terms in $\lambda\wedge''$ are of the form $\lambda x_A : A.M$. When Γ is such a basis and M is a term in $\lambda\wedge''$, then we write $\Gamma \vdash_{\lambda\wedge''} M : A$ for $\Gamma \vdash_{\lambda\wedge'} M : A$. The following lemma guarantees the reducibility of the emptiness problem of the set $\{M \text{ in } \beta\text{-nf} \mid \Gamma \vdash_{\lambda\wedge'} M : A\}$ to the same problem for $\lambda\wedge''$.

Lemma 5.2 Let Γ be a basis of the system $\lambda\wedge'$ and A be a type. Then

$$\exists M \text{ in } \beta\text{-nf } (\Gamma \vdash_{\lambda\wedge'} M : A) \Longleftrightarrow \exists M \text{ in } \beta\text{-nf } (\varphi(\Gamma) \vdash_{\lambda\wedge''} M : A)$$

where $\varphi(\Gamma) = \{x_B : B \mid (y : B) \in \Gamma \text{ for some } y\}$. □

For the sake of simplicity, in the system $\lambda\wedge''$ we consider the set of types $\mathrm{T}_\wedge$ modulo $=$. Also for simplicity, we will write $\{x_{A_1}, x_{A_2}, ..., x_{A_n}\}$ for the basis $\{x_{A_1} : A_1, x_{A_2} : A_2, ..., x_{A_n} : A_n\}$, and $\lambda x_A.M$ for the term $\lambda x_A : A.M$.

Note that in $\lambda\wedge''$, since bound variables indicate their types, α-conversion does *not* preserve types. For example, $\vdash_{\lambda\wedge''} \lambda x_A.x_A : A \to A$, but $\not\vdash_{\lambda\wedge''} \lambda x_\omega.x_\omega : A \to A$ unless $A = \omega$. Therefore in $\lambda\wedge''$ we do not identify α-convertible terms.

From Theorem 4.6, one can easily obtain the following characterization of judgements in $\lambda\wedge''$.

Corollary 5.3 If $A \neq \omega$, then

$$\Gamma \vdash_{\lambda\wedge''} \lambda x_{B_1} \cdots x_{B_p}.x_C M_1 M_2 \cdots M_q : A \Longleftrightarrow$$
for some m, $A_{i,j}$'s, A_i's, $n \geq 1$, $C_{i,j}$'s, C_i's and C',
$$\begin{cases} A^* \equiv \bigwedge_{i=1}^m (A_{i,1} \to \cdots \to A_{i,p} \to A_i), \\ x_C \in \Gamma \cup \{x_{B_1}, ..., x_{B_p}\}, \\ C^* \equiv \bigwedge_{i=1}^n (C_{i,1} \to \cdots \to C_{i,q} \to C_i) \wedge C', \\ \Gamma, x_{B_1}, ..., x_{B_p} \vdash_{\lambda\wedge''} M_j : \bigwedge_{i=1}^n C_{i,j} \ (j = 1, ..., q), \\ \bigvee_{i=1}^m A_{i,k} \leq B_k \ (k = 1, ..., p), \ \ \bigwedge_{i=1}^n C_i \leq \bigwedge_{i=1}^m A_i. \end{cases}$$ □

For a basis Γ of $\lambda\wedge''$ and a type A, we write

$$\mathbf{K}(\Gamma, A) =_{def} \{M \in \Lambda''_{\mathrm{nf}} \mid \Gamma \vdash_{\lambda\wedge''} M : A\}$$

where Λ''_{nf} stands for the set of β-nf's in the system $\lambda\wedge''$. Our goal is to get information for deciding whether $\mathbf{K}(\Gamma, A) = \emptyset$ or not. Let $S = \{(\Gamma, A) \mid \Gamma$ is a basis of $\lambda\wedge''$, A is a type$\}$, and $\mathbf{K}$ be the (infinite) sequence $\langle \mathbf{K}(s) \rangle_{s\in S}$ of sets $\mathbf{K}(s)$ ($\subseteq \Lambda''_{\mathrm{nf}}$). Then the content of Corollary 5.3 can be stated as $\mathbf{K}(s) = \mathbf{\Phi}_s(\mathbf{K})$ for $s = (\Gamma, A) \in S$ with $A \neq \omega$ where $\mathbf{\Phi}_s$ is the mapping from sequences $\mathbf{X} = \langle \mathbf{X}(s) \rangle_{s\in S}$ of subsets of Λ''_{nf} to subsets of Λ''_{nf} defined by

$$\begin{aligned}\mathbf{\Phi}_s(\mathbf{X}) =_{def} \{\lambda x_{B_1} \cdots x_{B_p}.x_C M_1 M_2 \cdots M_q \mid \\ & A^* \equiv \textstyle\bigwedge_{i=1}^m (A_{i,1} \to \cdots \to A_{i,p} \to A_i), \\ & x_C \in \Gamma \cup \{x_{B_1}, \dots, x_{B_p}\}, \\ & C^* \equiv \textstyle\bigwedge_{i=1}^n (C_{i,1} \to \cdots \to C_{i,q} \to C_i) \wedge C', \\ & M_j \in \mathbf{X}(\Gamma \cup \{x_{B_1}, \dots, x_{B_p}\}, \textstyle\bigwedge_{i=1}^n C_{i,j})\ (j = 1, \dots, q), \\ & \textstyle\bigvee_{i=1}^m A_{i,k} \leq B_k\ \ (k = 1, \dots, p),\ \textstyle\bigwedge_{i=1}^n C_i \leq \textstyle\bigwedge_{i=1}^m A_i \\ & \text{for some } m, A_{i,k}\text{'s}, A_i\text{'s}, n \geq 1, C_{i,j}\text{'s}, C_i\text{'s and } C'\}.\end{aligned}$$

We extend the definition of $\mathbf{\Phi}_s$ by $\mathbf{\Phi}_s(\mathbf{X}) =_{def} \Lambda''_{\mathrm{nf}}$ for $s = (\Gamma, \omega) \in S$ so that $\mathbf{K}(s) = \mathbf{\Phi}_s(\mathbf{K})$ holds for each $s \in S$. Then we know that the sequence $\mathbf{K} = \langle \mathbf{K}(s) \rangle_{s\in S}$ is the unique solution of the simultaneous equation $\mathbf{X} = \mathbf{\Phi}(\mathbf{X})$ where $\mathbf{\Phi}(\mathbf{X})$ stands for the sequence $\langle \mathbf{\Phi}_s(\mathbf{X}) \rangle_{s\in S}$ (cf. [TAH 94] Proposition 1.1). Moreover, when we write $\mathbf{K}_n(s) = \{M \in \mathbf{K}(s) \mid \mathrm{height}(M) \leq n\}$[4] ($n \geq 0$, $s \in S$) and $\mathbf{K}_n = \langle \mathbf{K}_n(s) \rangle_{s\in S}$, we have $\mathbf{K}_0 = \boldsymbol{\emptyset}$ (the sequence $\langle \emptyset \rangle_{s\in S}$ of empty set), $\mathbf{K}_{n+1} = \mathbf{K}_n \cup \mathbf{\Phi}(\mathbf{K}_n)$, and $\mathbf{K} = \bigcup_{n=0}^\infty \mathbf{K}_n$ (i.e., $\mathbf{K}(s) = \bigcup_{n=0}^\infty \mathbf{K}_n(s)$ for each $s \in S$).

We are to decide the emptiness of components $\mathbf{K}(s)$ of $\mathbf{K}$ based on the description $\mathbf{K} = \mathbf{\Phi}(\mathbf{K})$, but to do so directly seems difficult. So we consider a slightly different mapping $\mathbf{\Psi}$, and reduce our problem to the emptiness problem of the (unique) fixed point of $\mathbf{\Psi}$.

For each sequence $\mathbf{X} = \langle \mathbf{X}(s) \rangle_{s\in S}$ of subsets of Λ''_{nf}, let

$$\begin{aligned}\mathbf{\Psi}_s(\mathbf{X}) =_{def} \{\lambda x_{B_1} \cdots x_{B_p}.x_C M_1 M_2 \cdots M_q \mid \\ & A^* \equiv \textstyle\bigwedge_{i=1}^m (A_{i,1} \to \cdots \to A_{i,p} \to A_i), \\ & x_C \in \Gamma \cup \{x_{B_1}, \dots, x_{B_p}\}, \\ & C^* \equiv \textstyle\bigwedge_{i=1}^n (C_{i,1} \to \cdots \to C_{i,q} \to C_i) \wedge C', \\ & M_j \in \mathbf{X}(\Gamma \cup \{x_{B_1}, \dots, x_{B_p}\}, \textstyle\bigwedge_{i=1}^n C_{i,j})\ (j = 1, \dots, q), \\ & \textstyle\bigvee_{i=1}^m A_{i,k} = B_k\ \ (k = 1, \dots, p),\ \textstyle\bigwedge_{i=1}^n C_i \leq \textstyle\bigwedge_{i=1}^m A_i \\ & \text{for some } m, A_{i,k}\text{'s}, A_i\text{'s}, n \geq 1, C_{i,j}\text{'s}, C_i\text{'s and } C'\} \\ & \qquad \text{if } s = (\Gamma, A) \in S \text{ and } A \neq \omega; \\ \mathbf{\Psi}_s(\mathbf{X}) =_{def} \{x_\omega\} \quad & \qquad \text{if } s = (\Gamma, \omega) \in S.\end{aligned}$$

The definition of $\mathbf{\Psi}_s$ for $s = (\Gamma, A)$ with $A \neq \omega$ is same as $\mathbf{\Phi}_s$ except the condition for bound variables x_{B_k}; the types B_k of bound variables vary under the condition $\bigvee_{i=1}^m A_{i,k} \leq B_k$ in $\mathbf{\Phi}_s$, while $\bigvee_{i=1}^m A_{i,k} = B_k$ in $\mathbf{\Psi}_s$.

[4] height(M) means the height of Böhm tree of M.

As in the case of $\mathbf{\Phi}$, we see that the mapping $\mathbf{\Psi}$ has a unique fixed point, which we write $\mathbf{L} = \langle \mathbf{L}(s) \rangle_{s \in S}$. Then as before we have $\mathbf{L} = \bigcup_{n=0}^{\infty} \mathbf{L}_n$ where $\mathbf{L}_0 = \emptyset$ and $\mathbf{L}_{n+1} = \mathbf{L}_n \cup \mathbf{\Psi}(\mathbf{L}_n)$ $(n = 0, 1, ...)$. Between the fixed points $\mathbf{K}$ of $\mathbf{\Phi}$ and $\mathbf{L}$ of $\mathbf{\Psi}$, we can observe the following relations.

Lemma 5.4 $\mathbf{L} \subseteq \mathbf{K}$ (i.e., $\mathbf{L}(s) \subseteq \mathbf{K}(s)$ for each $s \in S$).
Proof. By definition of $\mathbf{\Psi}$, clearly $\mathbf{\Psi}(\mathbf{X}) \subseteq \mathbf{\Phi}(\mathbf{X})$ for each sequence $\mathbf{X}$. On the other hand, since $\mathbf{\Phi}$ is a monotone mapping (i.e., $\mathbf{\Phi}(\mathbf{X}) \subseteq \mathbf{\Phi}(\mathbf{X}')$ if $\mathbf{X} \subseteq \mathbf{X}'$), if $\mathbf{L}_n \subseteq \mathbf{K}_n$ then $\mathbf{L}_{n+1} = \mathbf{L}_n \cup \mathbf{\Psi}(\mathbf{L}_n) \subseteq \mathbf{L}_n \cup \mathbf{\Phi}(\mathbf{L}_n) \subseteq \mathbf{K}_n \cup \mathbf{\Phi}(\mathbf{K}_n) = \mathbf{K}_{n+1}$. Thus we know by induction that $\mathbf{L}_n \subseteq \mathbf{K}_n$ for each n. □

Lemma 5.5 For each $s \in S$, if $\mathbf{K}(s) \neq \emptyset$ then $\mathbf{L}(s) \neq \emptyset$.
Proof. We can verify a stronger statement; from any $M \in \mathbf{K}(s)$ a term $N \in \mathbf{L}(s)$ can be constructed by changing bound variables x_B in M to some $x_{B'}$ with $B' \leq B$ and by replacing some subterms with x_ω. The details of the proof are omitted. □

Corollary 5.6 $\forall s \in S\ (\mathbf{K}(s) = \emptyset \Longleftrightarrow \mathbf{L}(s) = \emptyset)$. □

Our problems are now reduced to the emptiness problem of $\mathbf{L}(\Gamma, A)$ for given $(\Gamma, A) \in S$. To solve the latter problem, we will use a result in formal language theory; we show that $\mathbf{L}(\Gamma, A)$ is a context-free language (over an alphabet consisting of variables, $\lambda, ., (,)$), and that a context-free grammar generating $\mathbf{L}(\Gamma, A)$ can be constructed from Γ and A. Then we can apply a well-known algorithm in formal language theory to see $\mathbf{L}(\Gamma, A) = \emptyset$ or not.

For the sake of simplicity, in the sequel we will write Γ for $\mathrm{Pred}(\Gamma) =_{def} \{B \mid x_B \in \Gamma\}$. Note that in the system $\lambda\wedge''$ the mapping Pred is a one-to-one correspondence between bases and finite sets of types.

Definition 5.7 Let $s = (\Gamma, A) \in S$. We define three sets T_s ($\subseteq \{x_B \mid B \in \mathrm{T}_\wedge\}$[5] $\cup \{\lambda, ., (,)\}$) of terminal symbols, N_s ($\subseteq S$) of nonterminal symbols, and R_s ($\subseteq \{\varphi \leadsto \varphi_1\varphi_2 \cdots \varphi_m \mid \varphi \in N_s, m \geq 1, \varphi_1, \ldots, \varphi_m \in T_s \cup N_s\}$) of production rules, recursively:

1. $\{x_C \mid C \in \Gamma\} \cup \{\lambda, ., (,)\} \subseteq T_s$ and $s \in N_s$.
2. If $s' = (\Gamma', \omega) \in N_s$, then $x_\omega \in T_s$ and $(s' \leadsto x_\omega) \in R_s$.
3. If $s' = (\Gamma', A') \in N_s$, $A' \neq \omega$, and

$$C \in \Gamma' \cup \{B_1, \ldots, B_p\},$$
$$C^* \equiv \textstyle\bigwedge_{i=1}^{n}(C_{i,1} \to \cdots \to C_{i,q} \to C_i) \wedge C',$$
$$s_j = (\Gamma' \cup \{B_1, \ldots, B_p\}, \textstyle\bigwedge_{i=1}^{n} C_{i,j}) \quad (j = 1, \ldots, q),$$
$$A'^* \equiv \textstyle\bigwedge_{i=1}^{m}(A'_{i,1} \to \cdots \to A'_{i,p} \to A'_i),$$
$$\textstyle\bigwedge_{i=1}^{n} C_i \leq \bigwedge_{i=1}^{m} A'_i, \quad \bigvee_{i=1}^{m} A'_{i,k} = B_k \quad (k = 1, \ldots, p),$$

then

$$x_{B_1}, \ldots, x_{B_p} \in T_s,\ s_1, \ldots, s_q \in N_s, \text{ and}$$
$$(s' \leadsto \lambda x_{B_1} \cdots x_{B_p}.x_C s_1 s_2 \cdots s_q) \in R_s. \quad \square$$

[5] Recall that in $\lambda\wedge''$ we consider types modulo $=$.

Roughly speaking, N_s is the set of s''s in S such that $\mathbf{X}(s')$ appears in the 'unfolding' of $\mathbf{X}(s) = \mathbf{\Psi}_s(\mathbf{X})$, and R_s consists of production rules of the form $s' \leadsto \lambda x_{B_1} \cdots x_{B_p}.x_C s_1 s_2 \cdots s_q$ where $s' \in N_s$, $\lambda x_{B_1} \cdots x_{B_p}.x_C M_1 M_2 \cdots M_q \in \mathbf{\Psi}_{s'}(\mathbf{X})$, and $M_j \in \mathbf{X}(s_j)$ $(j = 1, \ldots, q)$.

If the sets T_s, N_s and R_s are shown to be finite, then $G_s = (T_s, N_s, R_s, s)$ is a context-free grammar, and from the construction it clearly generates the set $\mathbf{L}(s)$.

In order to see the finiteness of sets T_s, N_s and R_s, it suffices to prove the finiteness of N_s. This is because for each $s' \in N_s$ only a finite number of production rules with s' in the lefthand sides are introduced in R_s, and the set T_s of terminal symbols in these rules is finite.

Theorem 5.8 The set N_s is finite for each $s \in S$.
Proof. See Appendix B. □

Theorem 5.9 For any Γ and A, whether $\mathbf{L}(\Gamma, A) = \emptyset$ or not is decidable.
Proof. Let $s = (\Gamma, A)$. First, following Definition 5.7 we enumerate members of sets T_s, N_s and R_s. At some point a same member of N_s appears twice by Theorem 5.8. Then we know all the elements of T_s, N_s and R_s, and the set $\mathbf{L}(s)$ can be generated by the context-free grammar $G_s = (T_s, N_s, R_s, s)$. It is well-known (cf. [HU 79]) that the context-free language $\mathbf{L}(s)$ generated by G_s is empty if and only if $\{M \in \mathbf{L}(s) \mid \text{height}(M) \leq n+1\} = \emptyset$ where n is the number of elements of N_s. Thus the emptiness of $\mathbf{L}(s)$ is decidable. □

Corollary 5.10 The inhabitation problems for $\lambda\wedge-(\wedge\mathrm{I})$ and for $\lambda\wedge'$ are decidable.
Proof. By Theorem 5.1, Lemma 5.2, Corollary 5.6 and Theorem 5.9. □

Appendix A. (Proof of Theorem 3.2)

In order to prove Theorem 3.2, we define *depth* and *length* of types, as follows: If $A^* \equiv \bigwedge_{i=1}^{m}(A_{i,1} \to \cdots \to A_{i,p_i} \to a_i) \not\equiv \omega$, then $\mathrm{dp}(A) \doteq \max\{\mathrm{dp}(A_{i,j}) \mid i = 1, \ldots, m,\ j = 1, \ldots, p_i\} + 1$ and $\mathrm{lg}(A) = \max\{p_i \mid i = 1, \ldots, m\}$. If $A = \omega$, we define $\mathrm{dp}(A) = \mathrm{lg}(A) = 0$. We will write $\mathrm{ord}(A)$ for $(\mathrm{dp}(A), \mathrm{lg}(A))$, and $\leq$ for the lexicographic order on pairs of natural numbers. Note that $\mathrm{ord}(A) = 0 \iff A = \omega$.

Lemma A.1 (1) $\mathrm{ord}(A_i) \leq \mathrm{ord}(\bigwedge_{i \in I} A_i)$ if $A_i \in \mathsf{NT}^{\to}$ $(i \in I)$.
(2) $\mathrm{ord}(A_i) < \mathrm{ord}(A_1 \to A_2)$ $(i = 1, 2)$. □

When γ is a derivation in the system $\lambda\wedge'$ of the form

$$\dfrac{\begin{array}{cc}\vdots & \vdots \\ \Gamma \vdash \lambda x : A.M : C \to D & \Gamma \vdash N : C\end{array}}{\Gamma \vdash (\lambda x : A.M)N : D}\ (\to \mathrm{E})$$

we say γ is a *cut* represented by the term $(\lambda x : A.M)N$, and its *order* ($\mathrm{ord}(\gamma)$ in notation) is defined by $\mathrm{ord}(C \to D)$.

Lemma A.2 For any derivation π of $\Gamma \vdash_{\lambda\wedge'} M : A$, there exists a term N and a derivation π' of $\Gamma \vdash_{\lambda\wedge'} N : A$ such that $M \twoheadrightarrow_\beta N$ and any cut in π' is of order $(0,0)$.

Proof. Suppose π contains some cuts γ with $\mathrm{ord}(\gamma) \neq (0,0)$. Then among the cuts in π of maximum order, say θ, let

$$\begin{array}{c}
\begin{array}{cc} \vdots & \vdots\ \delta \\ \lambda x : A.M : C \to D & N : C \end{array} \\
\hline
(\lambda x : A.M)N : D \\
\vdots\ \zeta
\end{array}\ (\to \mathrm{E})$$

be the one which is represented by a shortest term, and call it γ. Without loss of generality, we may assume that the major premiss $\lambda x : A.M : C \to D$ of γ is derived from subderivations of the form

$$\begin{array}{c}
x : A \\
\vdots\ \delta_i \\
M : B_i \\
\hline
\lambda x : A.M : A \to B_i
\end{array}\ (\to \mathrm{I})' \qquad (i = 1, \ldots, m)$$

and some instances of (ω)-rule $\lambda x : A.M : \omega$ by means of $(\wedge\mathrm{I})$- and $(\leq)$-rules. In general, if a statement $M : E$ is derived from $M : E_i$ $(i = 1, \ldots, m)$ by applying only $(\wedge\mathrm{I})$- and $(\leq)$-rules, then $\bigwedge_{i=1}^m E_i \leq E$ holds. Therefore we have $\bigwedge_{i=1}^m (A \to B_i) \leq C \to D \neq \omega$ since $\mathrm{ord}(C \to D) = \mathrm{ord}(\gamma) \neq (0,0)$. Then by Lemma 2.6 (3) we know that there exists a nonempty subset I of $\{1, \ldots, m\}$ such that $C \leq A$ and $\bigwedge_{i \in I} B_i \leq D$. In this case, we replace the cut γ with the following derivation.

$$\begin{array}{c}
\begin{array}{c}
\begin{array}{cc} \vdots\ \delta & \\ N : C & C \leq A \end{array} \\
\hline
N : A
\end{array}\ (\leq) \\
\vdots\ \delta_i' \\
M[x := N] : B_i \\
\vdots\ (\wedge\mathrm{I}) \\
\begin{array}{c}
\begin{array}{cc} M[x := N] : \bigwedge_{i \in I} B_i & \bigwedge_{i \in I} B_i \leq D \end{array} \\
\hline
M[x := N] : D
\end{array}\ (\leq) \\
\vdots\ \zeta'
\end{array}$$

Note that this replacement may create new cuts in δ_i' and/or ζ', and moreover some of them may be of order θ or more. If so, we can replace such cuts once again with derivations which contain only cuts of order less than θ. Indeed, suppose there is a new cut γ' in δ_i' such that $\mathrm{ord}(\gamma') \geq \theta$. Then γ' must be of the form

$$\begin{array}{c}
\begin{array}{cc}
\begin{array}{c}
\begin{array}{c}
\begin{array}{cc} \vdots\ \delta & \\ N : C & C \leq A \end{array} \\
\hline
N : A
\end{array}\ (\leq) \\
\vdots\ (\wedge\mathrm{I}), (\leq) \\
N : D \to E
\end{array} &
\begin{array}{c} \vdots\ \varepsilon \\ L : D \end{array}
\end{array} \\
\hline
NL : E
\end{array}\ (\to \mathrm{E})$$

where N is an abstraction. Then we have $C \leq D \rightarrow E \neq \omega$, and hence by Lemma 2.7 we can find $m, n \geq 1$, E_i's, C_j's and C'_j's such that $C^* \equiv \bigwedge_{j=1}^{n}(C_j \rightarrow C'_j)$, $E^* \equiv \bigwedge_{i=1}^{m} E_i$, and $\forall i \in \{1,\ldots,m\}\ \exists j_i \in \{1,\ldots,n\}\ (D \leq C_{j_i}$ and $C'_{j_i} \leq E_i)$. In this case, we can replace γ' with the following derivation.

$$
\dfrac{
\dfrac{
\dfrac{
\dfrac{\overset{\vdots\ \delta}{N : C} \quad C \leq C_{j_i} \rightarrow C'_{j_i}}{N : C_{j_i} \rightarrow C'_{j_i}}\ (\leq)
\quad
\dfrac{\overset{\vdots\ \varepsilon}{L : D} \quad D \leq C_{j_i}}{L : C_{j_i}}\ (\leq)
}{NL : C'_{j_i}}\ (\rightarrow \mathrm{E})
\quad C'_{j_i} \leq E_i
}{NL : E_i}\ (\leq)
}{}
$$

$$
\dfrac{\overset{\vdots\ (\wedge\mathrm{I})}{NL : \bigwedge_{i=1}^{m} E_i} \qquad \bigwedge_{i=1}^{m} E_i \leq E}{NL : E}\ (\leq)
$$

Note that this replacement does not create any cut of order θ or more. By a similar technique, we can also eliminate cuts in ζ' of order θ or more.

By applying the process repeatedly, we can obtain a term N and a derivation π' which satisfy the lemma. □

Theorem A.3 (Theorem 3.2) $\Gamma \vdash_{\lambda\wedge'} M : A \Longrightarrow \exists P \sqsubseteq M\ (\Gamma \vdash_{\lambda\wedge'} P : A)$.
Proof. Suppose N and π' are obtained from a derivation π of $\Gamma \vdash_{\lambda\wedge'} M : A$ by applying Lemma A.2. In the derivation π', each statement occurence whose subject is a β-redex is introduced by (ω)-rule or by a cut of order $(0,0)$. Then by replacing these statements by $\perp : \omega$, we get a derivation of $\Gamma \vdash_{\lambda\wedge'} P : A$ where P is a finite approximation of M. □

Appendix B. (Proof of Theorem 5.8)

In order to prove the finiteness of N_s, we introduce some auxiliary notions.

Definition B.1 Suppose $A^* \equiv \bigwedge_{i=1}^{m}(A_{i,1} \rightarrow \cdots \rightarrow A_{i,p_i} \rightarrow a_i)$. Then we define two subsets $\mathrm{dom}_\wedge(A)$ and $\mathrm{dom}^\vee(A)$ of $\mathrm{T}_\wedge$, as follows:

$$\mathrm{dom}_\wedge(A) = \{\textstyle\bigwedge_{i \in I} A_{i,j} \mid \emptyset \neq I \subseteq \{1,\ldots,m\},\ 1 \leq j \leq p_i\ (i \in I)\ \},$$
$$\mathrm{dom}^\vee(A) = \{\textstyle\bigvee_{i=1}^{m} A_{i,j} \mid 1 \leq j \leq p_i\ (i = 1,\ldots,m)\ \}. \qquad \square$$

Note that both $\mathrm{dom}_\wedge(A)$ and $\mathrm{dom}^\vee(A)$ are finite for each type A. For a set T of types we write $\mathrm{dom}_\wedge(T)$ and $\mathrm{dom}^\vee(T)$ for $\bigcup_{A\in T} \mathrm{dom}_\wedge(A)$ and $\bigcup_{A \in T} \mathrm{dom}^\vee(A)$, respectively.

Definition B.2 For each $s \in S$, $\mathrm{dom}(s)$ ($\subseteq \mathrm{T}_\wedge$) is defined recursively, as follows:

1. $\Gamma \cup \{A\} \subseteq \mathrm{dom}(s)$.
2. $\mathrm{dom}_\wedge(\mathrm{dom}(s) \cup \mathrm{dom}^\vee(\mathrm{dom}(s)) \subseteq \mathrm{dom}(s)$. □

In order to see the finiteness of N_s, it suffices to show that

$$N_s \subseteq \mathcal{P}(\mathrm{dom}(s) \cup \mathrm{dom}^\vee(\mathrm{dom}(s))) \times \mathrm{dom}(s),$$

and that dom(s) is finite. First we prove the inclusion relation.

Lemma B.3 For each $s \in S$, $N_s \subseteq \mathcal{P}(\mathrm{dom}(s) \cup \mathrm{dom}^{\vee}(\mathrm{dom}(s))) \times \mathrm{dom}(s)$.
Proof. By induction on the structure of N_s. Let $\Delta = \mathrm{dom}(s) \cup \mathrm{dom}^{\vee}(\mathrm{dom}(s))$.
(case 1) If $s = (\Gamma, A)$, then $\Gamma \cup \{A\} \subseteq \mathrm{dom}(s) \subseteq \Delta$ by definition of dom(s). Thus s belongs to $\mathcal{P}(\Delta) \times \mathrm{dom}(s)$.
(case 2) Suppose $s' = (\Gamma', A') \in N_s$ with $A' \neq \omega$, and $s'' = (\Gamma'', A'')$ is added to N_s in the recursion step 3 of Definition 5.7 (i.e., $\mathbf{X}(\Gamma'', A'')$ appears in the definition of $\Psi_{s'}(\mathbf{X})$). Then $\Gamma'' = \Gamma' \cup \{B_1, \ldots, B_p\}$ where $A'^* = \bigwedge_{i=1}^m (A'_{i,1} \to \cdots \to A'_{i,p} \to A')$, $B_k = \bigvee_{i=1}^m A'_{i,k}$ ($\in \mathrm{dom}^{\vee}(A')$) $(k = 1, \ldots, p)$, and $A'' = \bigwedge_{i=1}^n C_{i,j}$ ($\in \mathrm{dom}_{\wedge}(C)$) where $C \in \Gamma''$, $C^* \equiv \bigwedge_{i=1}^n (C_{i,1} \to \cdots \to C_{i,q} \to C_i) \wedge C'$ and $1 \leq j \leq q$. Then for each k, since $B_k \in \mathrm{dom}^{\vee}(A')$ and $A' \in \mathrm{dom}(s)$ (by induction hypothesis), by definition of Δ we have $B_k \in \Delta$. This means that $\Gamma'' = \Gamma' \cup \{B_1, \ldots, B_p\} \subseteq \Delta$ because $\Gamma' \subseteq \Delta$ by induction hypothesis. On the other hand, since $C \in \Gamma'' \subseteq \Delta$, we have $\mathrm{dom}_{\wedge}(C) \subseteq \mathrm{dom}_{\wedge}(\Delta) \subseteq \mathrm{dom}(s)$ by definition of dom(s) and Δ. This proves $A'' \in \mathrm{dom}_{\wedge}(C) \subseteq \mathrm{dom}(s)$. Hence $(\Gamma'', A'') \in \mathcal{P}(\Delta) \times \mathrm{dom}(s)$. $\square$

Next, to see the finiteness of dom(s), we define the degree of types, as follows:

$$\deg(A) = \begin{cases} \max\{\deg(A_{i,1}) + \cdots + \deg(A_{i,p_i}) + p_i \mid i = 1, \ldots, m\} \\ \qquad\qquad \text{if } A^* \equiv \bigwedge_{i=1}^m (A_{i,1} \to \cdots \to A_{i,p_i} \to a_i) \not\equiv \omega, \\ 0 \qquad\qquad \text{if } A^* \equiv \omega. \end{cases}$$

In the following three lemmas, we assume that $A^* \equiv \bigwedge_{i=1}^m (A_{i,1} \to \cdots \to A_{i,p_i} \to a_i)$.

Lemma B.4 $\deg(A_{i,j}) < \deg(A)$.
Proof. $\deg(A_{i,j}) < \deg(A_{i,1} \to \cdots \to A_{i,p_i} \to a_i) \leq \deg(A^*) = \deg(A)$. $\square$

Lemma B.5 If $B \in \mathrm{dom}_{\wedge}(A)$, then $\deg(B) < \deg(A)$.
Proof. Since $B^* \equiv \bigwedge_{i \in I} A_{i,j}$ for some I and j, $\deg(B) \leq \max\{\deg(A_{i,j}) \mid 1 \leq i \leq m, 1 \leq j \leq p_i\} < \deg(A)$ by Lemma B.4. $\square$

Lemma B.6 If $C \in \mathrm{dom}_{\wedge}(\mathrm{dom}^{\vee}(A))$, then $\deg(C) < \deg(A)$.
Proof. Let us call each $A_{i,j}$ a *component* of A^*, and similarly for other normal types. Suppose $C \in \mathrm{dom}_{\wedge}(B)$ and $B \in \mathrm{dom}^{\vee}(A)$. Then by definition of $\mathrm{dom}_{\wedge}(B)$, C is an intersection of components of B^*. On the other hand, since $B = \bigvee_{i=1}^m A_{i,j}$ for some j $(\leq \min\{p_1, \ldots, p_m\})$, by Definition 2.9 each component of B^* is an intersection of components of components $A_{i,j}$ of A^*. Hence C is also an intersection of components of components of A^*. Therefore by Lemma B.4 we get $\deg(C) < \deg(A)$. $\square$

Corollary B.7 The set dom(s) is finite for each $s \in S$.
Proof. By definition, each element of dom(s) is obtained as B_i for some $i \geq 0$ where $B_0 \in \Gamma \cup \{A\}$, and $B_i \in \mathrm{dom}_{\wedge}(B_{i-1}) \cup \mathrm{dom}_{\wedge}(\mathrm{dom}^{\vee}(B_{i-1}))$ $(i = 1, 2, \ldots)$. Then it is clear that possible B_0's are finite, and for each B_{i-1} the set of B_i's satisfying above condition is also finite. Moreover by Lemmas B.5 and B.6 $\deg(B_i) < \deg(B_{i-1})$, which implies that there is no infinite sequence

$B_0, B_1, B_2, \ldots$. Then by König's lemma we know that the set dom(s) is finite. □

Theorem B.8 (Theorem 5.8) The set N_s is finite for each $s \in S$.
Proof. By Lemma B.3 and Corollary B.7. □

Acknowledgements

The authors are very grateful to Mariangiola Dezani-Ciancaglini for helpful discussions and comments about the subject of this paper.

References

[Bar 92] H.P. Barendregt, Lambda calculi with types, in: *Handbook of Logic in Computer Science vol.II*, ed. S. Abramsky et al., Oxford University Press, 1992.

[BCD 83] H.P. Barendregt, M. Coppo and M. Dezani-Ciancaglini, A filter lambda model and the completeness of type assignment, *J. Symbolic Logic* 48 (1983), pp.931-940.

[CC 90] F. Cardonne and M. Coppo, Two extensions of Curry's type inference system, in: *Logic and Computer Science* ed. P. Odifreddi, Academin Press, pp.19-75, 1990.

[DM 86] M. Dezani-Ciancaglini and I. Margaria, A characterisation of F-complete type assignments, *Theoretical Computer Science* 45 (1986), pp. 121-157.

[Hin 82] J.R. Hindley, The simple semantics for Coppo-Dezani-Sallé types, *Lecture Notes in Computer Science* 137 (1982), pp.212-226.

[HU 79] J.E. Hopcroft and J.D. Ullman, *Introduction to Automata Theory, Languages and Computation*. Addison-Wesley, 1979.

[Rey 88] J.C. Reynolds, Preliminary design of the Programming Language Forsythe, Report CMU-CS-88-159, Carnegie-Mellon University, 1988.

[TAH 94] M. Takahashi, Y. Akama and S. Hirokawa, Normal proofs and their grammar, Lecture Notes in Computer Science 789 (1994), pp.465-493.

[Urz 94] P. Urzyczyn, The emptiness problem for intersection types, *Proceedings of Logic in Computer Science*, IEEE, 1994.