Chapter 14

An Exemplary System

31.10.2006:581

There are several systems that assign intersection types to untyped lambda terms. These will be collectively denoted by $\lambda_\cap$. In this section we consider one particular system of this family, $\lambda_\cap^{\mathrm{BCD}}$ in order to outline the concepts and related properties. Definitions and the statement of theorems will be given, but no proofs. These can be found in the next chapters of Part III.

One motivation for the system presented comes from trying to modify the system $\lambda_\to$ in such a way that not only subject reduction, but also subject expansion holds. The problem of subject expansion is the following. Suppose $\vdash_{\lambda_\to} M : A$ and that $M' \twoheadrightarrow_{\boldsymbol{\beta\eta}} M$. Does one have $\vdash_{\lambda_\to} M' : A$? Let us focus on one $\boldsymbol{\beta}$-step. So let $M \equiv (\lambda x.P)Q$ be a redex and suppose

$$\vdash_{\lambda_\to} P[x := Q] : A. \tag{1}$$

Do we have $\vdash_{\lambda_\to} (\lambda x.P)Q : A$? It is tempting to reason as follows. By assumption (1) also Q must have a type, say B. Then $(\lambda x.P)$ has a type $B{\to}A$ and therefore $\vdash_{\lambda_\to} (\lambda x.P)Q : A$. The mistake is that in (1) there may be several occurrences of Q, say $Q_1 \equiv Q_2 \equiv \ldots \equiv Q_n$, having as types respectively $B_1, \ldots ,B_n$. It may be impossible to find a single type for all the occurrences of Q and this prevents us from finding a type for the redex. For example

$$\begin{array}{rcl} \vdash_{\lambda_\to} (\lambda x.\mathsf{I}(\mathsf{K}x)(\mathsf{I}x)) & : & A{\to}A, \\ \nvdash_{\lambda_\to} (\lambda xy.x(\mathsf{K}y)(xy))\mathsf{I} & : & A{\to}A. \end{array}$$

The system introduced in this chapter with intersection types assigned to untyped lambda terms remedies the situation. The idea is that if the several occurrences of Q have to have different types $B_1, \ldots ,B_n$, we give them all of these types:

$$\vdash Q : B_1 \cap \ldots \cap B_n,$$

implying that for all i one has $Q : B_i$. Then we have

$$\begin{array}{rcll} \vdash (\lambda x.P) & : & B_1 \cap \ldots \cap B_n{\to}A & \quad\text{and} \\ \vdash ((\lambda x.P)Q) & : & A. & \end{array}$$

There is, however, a second problem. In the $\lambda\mathsf{K}$-calculus, with its terms $\lambda x.P$ such that $x \notin \mathrm{FV}(P)$ there is the extra problem that Q may not be typable at all, as it may not occur in $P[x := Q]$! This is remedied by allowing $B_1 \cap \ldots \cap B_n$ also for $n = 0$ and writing this type as $\top$, to be considered as the universal type, i.e. assigned to all terms. Then in case $x \notin \mathrm{FV}(P)$ one has

$$\begin{aligned} \vdash (\lambda x.P) &: \top{\to}A \qquad \text{and} \\ \vdash ((\lambda x.P)Q) &: A. \end{aligned}$$

This is the motivation to introduce a $\leq$ relation on types with largest element $\top$ and intersections such that $A \cap B \leq A$, $A \cap B \leq B$ and the extension of the type assignment by the sub-sumption rule $\Gamma \vdash M : A,\ A \leq B \ \Rightarrow\ \Gamma \vdash M : B$. It has as consequence that terms like $\lambda x.xx$ get as type $((A{\to}B) \cap A){\to}B$, while $(\lambda x.xx)(\lambda x.xx)$ only gets $\top$ as type. Also we have subject conversion

$$\Gamma \vdash M : A \ \&\ M =_\beta N \ \Rightarrow\ \Gamma \vdash N : A.$$

This has as consequence that one can create a lambda model in which the meaning of a closed term consists of the collection of types it gets. In this way new lambda models will be obtained and new ways to study classical models as well.

The type assignement system $\lambda_\cap^{\mathrm{BCD}}$ will be introduced in Section 14.1 and the correspondig filter model in 14.2.

14.1. The system of type assignment $\lambda_\cap^{\mathrm{BCD}}$

A typical member of the family of intersection type assignment systems is $\lambda_\cap^{\mathrm{BCD}}$. This system is introduced in Barendregt et al. [1983] as an extension of the initial system in Coppo and Dezani-Ciancaglini [1980].

14.1.1. Definition. Let $\mathbb{A}$ be a set of type atoms.

(i) The *intersection type language* over $\mathbb{A}$, denoted by $\mathbb{T} = \mathbb{T}_\cap^{\mathbb{A}}$ is defined by the following abstract syntax.

$$\mathbb{T} = \mathbb{A} \mid \mathbb{T}{\to}\mathbb{T} \mid \mathbb{T} \cap \mathbb{T}$$

(ii) Write

$$\begin{aligned} \mathbb{A}_\infty &= \{\psi_0, \psi_1, \psi_2, \ldots\} \\ \mathbb{A}_\infty^\top &= \mathbb{A}_\infty \cup \{\top\}, \end{aligned}$$

where the type atom $\top \notin \mathbb{A}_\infty$ is considered as a constant.

Notation. (i) A, B, C, D, E range over arbitrary types. When writing intersection types we shall use the following convention: the constructor $\cap$ takes precedence over the constructor $\to$ and it associates to the right. For example

$$(A{\to}B{\to}C) \cap A{\to}B{\to}C \equiv ((A{\to}(B{\to}C)) \cap A){\to}(B{\to}C).$$

(ii) $\alpha, \beta, \ldots$ range over $\mathbb{A}$.

14.1.2. Remark. In Part III the set of syntactic types will be formed as above; for many of these systems the set $\mathbb{A}$ will be finite. In this Chapter, however, we take $\mathbb{A} = \mathbb{A}_\infty^\top$.

The following deductive system has as intention to introduce an appropriate pre-order on $\mathbb{T}$, compatible with the operator $\to$, such that $A \cap B$ is a greatest lower bound of A and B, for each A, B.

14.1.3. Definition (Intersection type preorder). On $\mathbb{T} = \mathbb{T}_\cap^{\mathbb{A}_\infty^\top}$ a binary relation $\leq$ '*is subtype of*' is defined by the following axioms and rules.

(refl)	$A \leq A$
(incl_L)	$A \cap B \leq A$
(incl_R)	$A \cap B \leq B$
(glb)	$\dfrac{C \leq A \quad C \leq B}{C \leq A \cap B}$
(trans)	$\dfrac{A \leq B \quad B \leq C}{A \leq C}$
$(\top)$	$A \leq \top$
$(\top\!\to)$	$\top \leq \top\!\to\!\top$
$(\to\cap)$	$(A\to B) \cap (A\to C) \leq A\to(B \cap C)$
$(\to)$	$\dfrac{A' \leq A \quad B \leq B'}{(A\to B) \leq (A'\to B')}$

14.1.4. Definition. The intersection type theory BCD is the set of all judgements $A \leq B$ derivable from the axioms and rules in Definition 14.1.3. For $(A \leq B) \in \mathrm{BCD}$ we write $A \leq_{\mathrm{BCD}} B$ or $\vdash_{\mathrm{BCD}} A \leq B$ (or often just $A \leq B$).

14.1.5. Remark. All systems in Part III have the first five axioms and rules of Definition 14.1.3. They differ in the extra axioms and rules and the set of constants.

14.1.6. Definition. Write $A =_{\mathrm{BCD}} B$ (or $A = B$) for $A \leq_{\mathrm{BCD}} B$ & $B \leq_{\mathrm{BCD}} A$. In BCD we usually work with $\mathbb{T}$ modulo $=_{\mathrm{BCD}}$. By rule $(\to)$ one has

$$A = A' \ \&\ B = B' \ \Rightarrow\ (A\to B) = (A'\to B').$$

Moreover, $A \cap B$ becomes *the* glb of A, B.

14.1.7. Definition. (i) A *basis* is a finite set of statements of the shape $x{:}B$, where $B \in \mathbb{T}$, with all variables distinct.

(ii) The type assignment system $\lambda_\cap^{\mathrm{BCD}}$ for deriving statements of the form $\Gamma \vdash M : A$ with Γ a basis, $M \in \Lambda$ (the set of untyped lambda terms) and $A \in \mathbb{T}$

is defined by the following axioms and rules.

(Ax)	$\Gamma \vdash x{:}A$	if $(x{:}A) \in \Gamma$
$(\to\mathrm{I})$	$\dfrac{\Gamma,\ x{:}A \vdash M : B}{\Gamma \vdash (\lambda x.M) : (A\to B)}$	
$(\to\mathrm{E})$	$\dfrac{\Gamma \vdash M : (A \to B) \quad \Gamma \vdash N : A}{\Gamma \vdash (MN) : B}$	
$(\cap\mathrm{I})$	$\dfrac{\Gamma \vdash M : A \quad \Gamma \vdash M : B}{\Gamma \vdash M : (A \cap B)}$	
$(\leq)$	$\dfrac{\Gamma \vdash M : A}{\Gamma \vdash M : B}$	if $A \leq_{\mathrm{BCD}} B$
$(\top\text{-universal})$	$\Gamma \vdash M : \top$	

(iii) We say that a term M is *typable* from a given basis Γ, if there is a type $A \in \mathbb{T}$ such that the judgement $\Gamma \vdash M : A$ is derivable in $\lambda_\cap^{\mathrm{BCD}}$. In this case we write $\Gamma \vdash_{\cap\top}^{BCD} M : A$ or just $\Gamma \vdash M : A$, if there is little danger of confusion.

14.1.8. Remark. All systems of type assignment in Part III have the first five axioms and rules of Definition 14.1.7.

In the following Proposition we need the notions of admissible and derived rule. Let us first informally define these notions for the simple logical theory of propositional logic.

14.1.9. Definition. Let $\vdash$ denote provability in propositional logic. Consider the rule

$$\frac{\Gamma \vdash A}{\Gamma \vdash B} \qquad (R)$$

(i) R is called *admissible* if one has

$$\Gamma \vdash A \ \Rightarrow\ \Gamma \vdash B$$

(ii) R is called *derived* if one has

$$\Gamma \vdash A\to B$$

For example we have that

$$\frac{\Gamma \vdash A\to A\to B}{\Gamma \vdash A\to B}$$

is derived. Also that for propositional variables ϑ, ϱ

$$\frac{\vdash \vartheta}{\vdash \varrho}$$

is admissible, simply because $\vdash \vartheta$ does not hold, but not derived. A derived rule is always admissible and the example shows that the converse does not hold. If

$$\frac{\Gamma \vdash A}{\Gamma \vdash B}$$

is a derived rule, then for all $\Gamma' \supseteq \Gamma$ one has that

$$\frac{\Gamma' \vdash A}{\Gamma' \vdash B}$$

is also derived. Hence derived rules are closed under theory extension.

We will only be concerned with admissible and derived rules for theories of type assignment.

14.1.10. PROPOSITION. (i) *Notice that the rules* $(\cap E)$

$$\frac{\Gamma \vdash M : (A\cap B)}{\Gamma \vdash M : A} \qquad \frac{\Gamma \vdash M : (A\cap B)}{\Gamma \vdash M : B}$$

are derived in $\lambda_\cap^{\rm BCD}$.

(ii) *The following rules are admissible in the intersection type assignment system* $\lambda_\cap^{\rm BCD}$.

(*weakening*)	$\dfrac{\Gamma \vdash M : A \quad x\notin \Gamma}{\Gamma, x{:}B \vdash M : A}$
(*strengthening*)	$\dfrac{\Gamma, x{:}B \vdash M : A \quad x\notin FV(M)}{\Gamma \vdash M : A}$
(*cut*)	$\dfrac{\Gamma, x{:}B \vdash M : A \quad \Gamma \vdash N : B}{\Gamma \vdash (M[x:=N]) : A}$
($\leq$-L)	$\dfrac{\Gamma, x{:}B \vdash M : A \quad C\leq B}{\Gamma, x{:}C \vdash M : A}$
($\to$L)	$\dfrac{\Gamma, y{:}B \vdash M : A \quad \Gamma \vdash N : C \quad x\notin\Gamma}{\Gamma, x{:}(C\to B) \vdash (M[y:=xN]) : A}$
($\cap$L)	$\dfrac{\Gamma, x{:}A \vdash M : B}{\Gamma, x{:}(A\cap C) \vdash M : B}$

14.1.11. THEOREM. *In (i) assume* $A \neq \top$. *Then*

(i) $\Gamma \vdash x : A \iff \exists B\in\mathbb{T}.[(x{:}B\in\Gamma \;\&\; B\leq A]$.

(ii) $\Gamma \vdash (MN) : A \iff \exists B\in\mathbb{T}.[\Gamma\vdash M : (B\to A) \;\&\; \Gamma\vdash N : B]$.

(iii) $\Gamma \vdash \lambda x.M : A \iff \exists n{>}0\,\exists B_1,\dots,B_n, C_1,\dots,C_n\in\mathbb{T}$
$\forall i\in\{1,\dots,n\}.[\Gamma, x{:}B_i \vdash M : C_i \;\&$
$(B_1\to C_1)\cap\dots\cap(B_n\to C_n)\leq A]$.

(iv) $\Gamma \vdash \lambda x.M : B\to C \iff \Gamma, x{:}B\vdash M : C$.

14.1.12. Definition. Let R be a notion of reduction. We introduce the following rules:

$$(R\text{-}red)\quad \frac{\Gamma \vdash M : A \quad M \to_R N}{\Gamma \vdash N : A}$$

$$(R\text{-}exp)\quad \frac{\Gamma \vdash M : A \quad M \leftarrow_R N}{\Gamma \vdash N : A}$$

14.1.13. Proposition. *The rules (β-red), (β-exp) and (η-red) are admissible in $\lambda_\cap^{\mathrm{BCD}}$. The rule ($\eta$-exp) is not.*

The following result characterizes notions related to normalization in terms of type assignment in the system $\lambda_\cap^{\mathrm{BCD}}$. The notation $\top \notin A$ means that $\top$ does not occur in A.

14.1.14. Theorem. *Let $M \in \Lambda^{\emptyset}$.*

(i) *M has a head normal form* $\Leftrightarrow$ $\exists A \in \mathbb{T}.[A \neq_{\mathrm{BCD}} \top \;\&\; \vdash M : A]$.

(ii) *M has a normal form* $\Leftrightarrow$ $\exists A \in \mathbb{T}.[\top \notin A \;\&\; \vdash M : A]$.

Let M be a lambda term. For the notion 'approximant of M', see Barendregt [1984]. These are roughly obtained from the Böhm tree $\mathrm{BT}(M)$ of M by cutting of branches and replacing these by a new symbol $\bot$. The set of approximants of M is denoted by $\mathcal{A}(M)$. We have e.g. for the fixed-point combinator Y

$$\mathcal{A}(\mathsf{Y}) = \{\bot\} \cup \{\lambda f.f^n\bot \mid n{>}0\}.$$

Approximants are being typed by letting the typing rules be valid for approximants. For example one has

$$\begin{array}{l}\vdash \bot : \top\\ \vdash \lambda f.f\bot : (\top{\to}A_1){\to}A_1\\ \vdash \lambda f.f(f\bot) : (\top{\to}A_1) \cap (A_1{\to}A_2){\to}A_2\\ \ldots\\ \vdash \lambda f.f^n\bot : (\top{\to}A_1) \cap (A_1{\to}A_2) \cap \ldots \cap (A_{n-1}{\to}A_n){\to}A_n\\ \ldots\end{array}$$

The set of types of a term M coincides with the union of the sets of types of its approximants $P \in \mathcal{A}(M)$. This will give an Approximation Theorem for the filter model of next section.

14.1.15. Theorem. *$\Gamma \vdash M : A \;\Leftrightarrow\; \exists P \in \mathcal{A}(M).\Gamma \vdash P : A$.*

For example since for all n $\lambda f.f^n\bot$ is an approximant of Y we have that all types of the shape $(\top{\to}A_1) \cap \ldots \cap (A_{n-1}{\to}A_n){\to}A_n$ can be derived for Y.

Finally the question whether an intersection type is inhabited is undecidable.

14.1.16. Theorem. *The set $\{A \in \mathbb{T} \mid \exists M \in \Lambda^{\emptyset} \vdash M : A\}$ is undecidable.*

14.2. The filter model

14.2.1. Definition. (i) A *complete lattice* $(\mathcal{D}, \sqsubseteq)$ is a partial order which has arbitrary least upper bounds (sup's) (and hence has arbitrary inf's).

(ii) A subset $Z \subseteq \mathcal{D}$ is *directed* if $Z \neq \emptyset$ and

$$\forall x, y \in Z \exists z \in Z. x, y \sqsubseteq z.$$

(iii) An element $c \in \mathcal{D}$ is *compact* (in the literature also called *finite*) if for each directed $Z \subseteq \mathcal{D}$ one has

$$c \sqsubseteq \bigsqcup Z \;\Rightarrow\; \exists z \in Z. c \sqsubseteq z.$$

Let $\mathcal{K}(\mathcal{D})$ denote the set of compact elements of $\mathcal{D}$.

(iv) A complete lattice is ω-*algebraic* if $\mathcal{K}(\mathcal{D})$ is countable, and for each $d \in \mathcal{D}$, the set $\mathcal{K}(d) = \{c \in \mathcal{K}(\mathcal{D}) \mid c \sqsubseteq d\}$ is directed and $d = \bigsqcup \mathcal{K}(d)$.

(v) Let $(\mathcal{D}, \sqsubseteq)$ be an ω-*algebraic* complete lattice. The Scott topology on $\mathcal{D}$ contains as open sets the $U \subseteq \mathcal{D}$ such that

(1) $d \in U \;\&\; d \sqsubseteq e \;\Rightarrow\; e \in U$;

(2) if $Z \subseteq \mathcal{D}$ is directed then $\bigsqcup Z \in U \;\Rightarrow\; \exists z \in Z. z \in U$.

(vi) If $\mathcal{D}, \mathcal{E}$ are ω-algebraic complete lattices, then $[\mathcal{D} \to \mathcal{E}]$ denotes the set of continuous maps from $\mathcal{D}$ to $\mathcal{E}$. This set can be ordered pointwise

$$f \sqsubseteq g \;\Leftrightarrow\; \forall d \in \mathcal{D}. f(d) \sqsubseteq g(d)$$

and $\langle [\mathcal{D} \to \mathcal{E}], \sqsubseteq \rangle$ is again an ω-algebraic lattice.

(vii) The category **ALG** is the category whose objects are the ω-algebraic complete lattices and whose morphisms are the (Scott) continuous functions.

14.2.2. Definition. (i) A *filter* over $\mathbb{T} = \mathbb{T}_{\cap}^{\mathbb{A}_\infty^\top}$ is a non-empty set $X \subseteq \mathbb{T}$ such that

(1) $A \in X \;\&\; A \leq B \;\Rightarrow\; B \in X$;

(2) $A, B \in X \;\Rightarrow\; (A \cap B) \in X$.

(ii) $\mathcal{F}$ denotes the set of filters over $\mathbb{T}$.

14.2.3. Definition. (i) If $X \subseteq \mathbb{T}$ is non-empty, then the filter *generated by* X, notation $\uparrow X$, is the least filter containing X. Note that

$$\uparrow X = \{A \mid \exists n \geq 1 \, \exists B_1 \ldots B_n \in X. B_1 \cap \ldots \cap B_n \leq A\}.$$

(ii) A *principal* filter is of the form $\uparrow\{A\}$ for some $A \in \mathbb{T}$. We shall denote this simply by $\uparrow A$. Note that $\uparrow A = \{B \mid A \leq B\}$.

14.2.4. Proposition. (i) $\mathcal{F} = \langle \mathcal{F}, \subseteq \rangle$ *is an* ω-*algebraic complete lattice.*

(ii) $\mathcal{F}$ *has as bottom element* $\uparrow\top$ *and as top element* $\mathbb{T}$.

(iii) *The compact elements of* $\mathcal{F}$ *are exactly the principal filters.*

14.2.5. Definition. Let $\mathcal{D}$ be an ω-algebraic lattice and let

$$\begin{array}{rcl} F & : & \mathcal{D}\to[\mathcal{D}\to\mathcal{D}] \\ G & : & [\mathcal{D}\to\mathcal{D}]\to\mathcal{D} \end{array}$$

be Scott continuous. $\mathcal{D}$ is called a *reflexive* via F, G if $F \circ G = \mathtt{id}_{[\mathcal{D}\to\mathcal{D}]}$.

A reflexive element of **ALG** is also a λ-model in which the term interpretation is naturally defined as follows (see Barendregt [1984], Section 5.4).

14.2.6. Definition (Interpretation of terms). Let $\mathcal{D}$ be reflexive via F, G.

(i) A *term environment* in $\mathcal{D}$ is a map $\rho : \mathtt{Var}\to\mathcal{D}$.

(ii) If ρ is a term environment and $d \in \mathcal{D}$, then $\rho(x := d)$ is the term environment ρ' defined by

$$\begin{array}{rcll} \rho'(y) & = & \rho(y) & \text{if } y \not\equiv x; \\ \rho'(x) & = & d. & \end{array}$$

(iii) Given a term environment ρ, the interpretation $[\![\ ]\!]_\rho : \Lambda\to\mathcal{D}$ is defined as follows.

$$\begin{array}{rcl} [\![x]\!]^{\mathcal{D}}_\rho & = & \rho(x); \\ [\![MN]\!]^{\mathcal{D}}_\rho & = & F[\![M]\!]^{\mathcal{D}}_\rho[\![N]\!]^{\mathcal{D}}_\rho; \\ [\![\lambda x.M]\!]^{\mathcal{D}}_\rho & = & G(\boldsymbol{\lambda} d \in \mathcal{D}.[\![M]\!]^{\mathcal{D}}_{\rho(x:=d)}). \end{array}$$

(iv) The statement $M = N$, for M, N untyped lambda terms, is *true in* $\mathcal{D}$, notation $\mathcal{D} \models M = N$ iff

$$\forall \rho \in \mathsf{Env}_D.[\![M]\!]^{\mathcal{D}}_\rho = [\![N]\!]^{\mathcal{D}}_\rho.$$

14.2.7. Theorem. *Let $\mathcal{D}$ be reflexive via F, G. Then $\mathcal{D}$ is a λ-model, in particular for all $M, N \in \Lambda$*

$$\mathcal{D} \models (\lambda x.M)N = M[x := N].$$

14.2.8. Proposition. *Define maps $F : \mathcal{F}\to[\mathcal{F}\to\mathcal{F}]$ and $G : [\mathcal{F}\to\mathcal{F}]\to\mathcal{F}$ by*

$$\begin{array}{rcl} F(X)(Y) & = & \uparrow\{B \mid \exists A \in Y.(A\to B) \in X\} \\ G(f) & = & \uparrow\{A\to B \mid B \in f(\uparrow A)\}. \end{array}$$

Then $\mathcal{F}$ is reflexive via F, G. Therefore $\mathcal{F}$ is a λ-model.

An important property of the λ-model $\mathcal{F}$ is that the meaning of a term is the set of types which are deducible for it.

14.2.9. Theorem. *For all λ-terms M one has*

$$[\![M]\!]^{\mathcal{F}}_\rho = \{A \mid \exists \Gamma \models \rho.\Gamma \vdash M : A\},$$

where $\Gamma \models \rho$ iff for all $(x{:}B) \in \Gamma$ one has $B \in \rho(x)$.

Lastly we notice that all continous functions are representable.

14.2.10. THEOREM.

$$[\mathcal{F}{\to}\mathcal{F}] = \{f : \mathcal{F}{\to}\mathcal{F} \mid f \textit{ is representable}\},$$

where $f \in \mathcal{F}{\to}\mathcal{F}$ *is called* representable *iff for some* $X \in \mathcal{F}$ *one has*

$$\forall Y \in \mathcal{F}. f(Y) = F(X)(Y).$$

14.3. Completeness of type assignment

14.3.1. DEFINITION (Interpretation of types). Let $\mathcal{D}$ be reflexive via F, G and hence a λ-model. For $F(d)(e)$ we also write (as usual) $d \cdot e$.

(i) A *type environment* in $\mathcal{D}$ is a map $\xi : \mathbb{A}_\infty{\to}\mathcal{P}(\mathcal{D})$.

(ii) For $X, Y \in \mathcal{P}(\mathcal{D})$ define

$$X{\to}Y = \{d \in \mathcal{D} \mid d \cdot X \subseteq Y\} = \{d \in \mathcal{D} \mid \forall x \in X. d \cdot x \in Y\}.$$

(iii) Given a type environment ξ, the interpretation $[\![\ ]\!]_\xi : \mathbb{T}{\to}\mathcal{P}(\mathcal{D})$ is defined as follows.

$$\begin{array}{rcll} [\![\top]\!]^{\mathcal{D}}_\xi & = & \mathcal{D}; & \\ [\![\alpha]\!]^{\mathcal{D}}_\xi & = & \xi(\alpha), & \text{for } \alpha \in \mathbb{A}_\infty; \\ [\![A{\to}B]\!]^{\mathcal{D}}_\xi & = & [\![A]\!]^{\mathcal{D}}_\xi {\to} [\![B]\!]^{\mathcal{D}}_\xi; & \\ [\![A \cap B]\!]^{\mathcal{D}}_\xi & = & [\![A]\!]^{\mathcal{D}}_\xi \cap [\![B]\!]^{\mathcal{D}}_\xi. & \end{array}$$

14.3.2. DEFINITION (Satisfaction). (i) Given a λ-model $\mathcal{D}$, a term environment ρ and a type environment ξ one defines the following.

$$\begin{array}{rcll} \mathcal{D}, \rho, \xi \models M : A & \Leftrightarrow & [\![M]\!]^{\mathcal{D}}_\rho \in [\![A]\!]^{\mathcal{D}}_\xi. & \\ \mathcal{D}, \rho, \xi \models \Gamma & \Leftrightarrow & \mathcal{D}, \rho, \xi \models x : B, & \text{for all } (x{:}B) \in \Gamma. \end{array}$$

(ii) $\Gamma \models M : A \iff \forall \mathcal{D}, \rho, \xi. [\mathcal{D}, \rho, \xi \models \Gamma \;\Rightarrow\; \rho, \xi \models M : A]$.

14.3.3. THEOREM (Soundness).

$$\Gamma \vdash M : A \;\Rightarrow\; \Gamma \models M : A.$$

14.3.4. THEOREM (Completeness).

$$\Gamma \models M : A \;\Rightarrow\; \Gamma \vdash M : A.$$

The completeness proof is an application of the λ-model $\mathcal{F}$, see Barendregt et al. [1983].