A Complete Characterization of Complete Intersection-Type Theories *

Extended Abstract

M. DEZANI-CIANCAGLINI
Università di Torino, Italy
F. HONSELL
Università di Udine, Italy
F. ALESSI
Università di Udine, Italy

Abstract

We characterize those *intersection-type theories* which yield complete *intersection-type assignment systems* for λ-calculi, with respect to the three canonical set-theoretical semantics for intersection-types: the inference semantics, the simple semantics and the F-semantics.

Keywords

Lambda Calculus, Intersection Types, Semantic Completeness, Filter Structures.

1 Introduction

Intersection-types disciplines originated in [6] to overcome the limitations of Curry's type assignment system and to provide a characterization of *strongly normalizing terms* of the λ-calculus. But very early on, the issue of *completeness* became crucial. Intersection-type theories and filter λ-models have been introduced, in [5], precisely to achieve the completeness for the type assignment system $\lambda\cap_{\Omega}^{\mathcal{BCD}}$, with respect to Scott's simple semantics. And this result, together with the conservativity of $\lambda\cap_{\Omega}^{\mathcal{BCD}}$, with respect to Curry's simple types, was used in [5] to prove Scott's conjecture concerning the completeness of the set-theoretic semantics for simple types.

The number of type theories of interest in the literature has grown considerably over the years (e.g. [9, 8, 17, 11, 2, 20, 16], etc.), especially in connection with the study of domain models for λ-calculi in the perspective of Abramsky's "domain theory in logical form" [1]. Furthermore new semantics have been proposed for intersection-types [14].

The problem of characterizing syntactically the sound and adequate (complete) intersection-type theories, with respect to the various set-theoretic semantics, appears therefore rather natural. Moreover, we feel that the very existence

*Partially supported by MURST Cofin '99 TOSCA Project and FGV '98.

of completeness results with respect to set-theoretic semantics, such as the one in [5], is probably one of the most significant features of intersection-types.

In this paper we solve completely the characterization problem as far as the three canonical *set-theoretical* semantics for intersection-types: the inference semantics, the simple semantics [22] and the F-semantics [25]. These are the semantics which arise by interpreting types as subsets of applicative structures, and by taking as interpretation of the *intersection constructor*, $\cap$, set-theoretic inclusion, and by taking the interpretation of the *arrow constructor*, $\to$, *à la Scott* as a *logical predicate*, with respect to either *any possible functionality set*, or the *largest* one, or the *least* one.

More precisely, the *simple semantics* of types associates to each arrow type $A \to B$ the set of elements which applied to an arbitrary element in the interpretation of A return an element in the interpretation of B.

As Scott has pointed out in [25], however, the key to a lambda model is the set of elements in the domain which are *canonical representatives* of functions, i.e. the elements which are meanings of terms starting with an initial abstraction. The F-semantics of types takes therefore as meaning of an arrow type only those elements which behave as expected with respect to application *and* which are also canonical representatives of functions.

The inference semantics is the counterpart of the inference semantics for polymorphic types introduced in [19], generalized to suitable applicative structures with an interpretation function, called λ-applicative structures. Here the interpretation of arrows is taken with respect to an arbitrary set which includes the canonical representatives of functions.

The results in this paper strengthen and generalize significantly all earlier results in the literature, to our knowledge, in at least three respects. First of all the inference semantics had not been considered before. Secondly the characterizations are all given just in terms of simple closure conditions on the *preorder relation* , $\leq$, on the types, rather than on the typing judgments themselves, as had been done earlier [10]. The task of checking the condition is made therefore considerably more tractable. Lastly we do not restrict attention just to λ-models, but to the more general class of λ-applicative structures. Thus we allow also for the treatment of models of restricted λ-calculi, and most notably models of Plotkin's call-by-value λ_v-calculus, and models of the λ-**I**-**N**-calculus of [16]. Nevertheless the characterizations we give can be tailored just to the case of λ-models.

The paper is organized as follows. In Section 2 we introduce intersection-type theories and various kinds of type assignment systems. In Section 3 we introduce the basic semantical structures, with respect to which we shall discuss soundness and completeness of intersection-type theories. In Section 4 we study filter structures and state the crucial property satisfied by the interpretation function over them. Section 5 is the main section of the paper. After introducing the notions of type interpretation domain and semantic satisfiability for the three semantics under consideration, we give the characterization results. Finally in Section 6 we

(Ω)	$A \leq \Omega$
(ν)	$A \to B \leq \nu$
$(\Omega\text{-}\eta)$	$\Omega \leq \Omega \to \Omega$
$(\Omega\text{-}lazy)$	$A \to B \leq \Omega \to \Omega$
$(\to\text{-}\cap)$	$(A \to B) \cap (A \to C) \leq A \to B \cap C$
(η)	$\dfrac{A' \leq A \quad B \leq B'}{A \to B \leq A' \to B'}$

Figure 1: Some special purpose Axioms and Rules concerning $\leq$.

$\mathsf{C}^{\mathcal{B}a}$	$=$	C_∞	$\mathcal{B}a$	$=$	$\{(\to\text{-}\cap), (\eta)\}$
$\mathsf{C}^{\mathcal{EHR}}$	$=$	$\{\nu\}$	$\mathcal{EHR}$	$=$	$\mathcal{B}a \cup \{(\nu)\}$
$\mathsf{C}^{\mathcal{AO}}$	$=$	$\{\Omega\}$	$\mathcal{AO}$	$=$	$\mathcal{B}a \cup \{(\Omega), (\Omega\text{-}lazy)\}$
$\mathsf{C}^{\mathcal{BCD}}$	$=$	$\{\Omega\} \cup \mathsf{C}_\infty$	$\mathcal{BCD}$	$=$	$\mathcal{B}a \cup \{(\Omega), (\Omega\text{-}\eta)\}$

Figure 2: Type Theories: atoms, axioms and rules.

discuss related results and give some final remarks.

2 Intersection-types and type assignment systems

Intersection-types are syntactical objects which are built inductively by closing a given set C of *type atoms* (constants) under the *function type* constructor $\to$ and the *intersection* type constructor $\cap$.

Definition 1 (Intersection-type Languages) *An* intersection-type language *over* C*, denoted by* $\mathsf{T} = \mathsf{T}(\mathsf{C})$ *is defined by the following abstract syntax:*

$$\mathsf{T} = \mathsf{C} \mid \mathsf{T} \to \mathsf{T} \mid \mathsf{T} \cap \mathsf{T}.$$

Upper case Roman letters i.e. $A, B, \ldots$, will denote arbitrary types. In writing intersection-types we shall use the following convention: the constructor $\cap$ takes precedence over the constructor $\to$ and it associates to the right. Moreover $A^n \to B$ will be short for $\underbrace{A \to \cdots \to A}_{n} \to B$.

Much of the expressive power of intersection-type disciplines comes from the fact that types can be endowed with a *preorder relation*, $\leq$, which induces the structure of a meet semi-lattice with respect to $\cap$.

Definition 2 (Intersection-type Preorder) *Let* $\mathsf{T} = \mathsf{T}(\mathsf{C})$ *be an intersection-type language. An* intersection-type preorder *over* T *is a binary relation* $\leq$ *on* T *satisfying the following set* $\bigtriangledown^0$ *("nabla-zero") of axioms and rules:*

$$\begin{array}{llll} (\mathit{refl}) & A \leq A & (\mathit{idem}) & A \leq A \cap A \\[1ex] (\mathit{incl}_L) & A \cap B \leq A & (\mathit{incl}_R) & A \cap B \leq B \\[1ex] (\mathit{mon}) & \dfrac{A \leq A' \quad B \leq B'}{A \cap B \leq A' \cap B'} & (\mathit{trans}) & \dfrac{A \leq B \quad B \leq C}{A \leq C} \end{array}$$

We will write $A \sim B$ for $A \leq B$ and $B \leq A$.
Notice that associativity and commutativity of $\cap$ (modulo $\sim$) follow easily from the above axioms and rules. Being $\cap$ commutative and associative, we will write $\cap_{i \leq n} A_i$ for $A_1 \cap \ldots \cap A_n$. Similarly we will write $\cap_{i \in I} A_i$, where we convene that I denotes always a finite non-empty set.

Possibly effective, syntactical presentations of intersection-type preorders can be given using the notion of *intersection-type theory*. An intersection-type theory always includes the basic set of axioms and rules for $\leq$, $\bigtriangledown^0$, together with possibly other special purpose axioms and rules.

Definition 3 (Intersection-type Theories) *Let* $\mathsf{T} = \mathsf{T}(\mathsf{C})$ *be an intersection-type language, and let* $\bigtriangledown$ *be a collection of axioms and rules for deriving judgments of the shape* $A \leq B$*, with* $A, B \in \mathsf{T}$*. The* intersection-type theory $\Sigma(\mathsf{C}, \bigtriangledown)$ *is the set of all judgments* $A \leq B$ *derivable from the axioms and rules in* $\bigtriangledown^0 \cup \bigtriangledown$*.*

When we consider the intersection-type theory $\Sigma(\mathsf{C}, \bigtriangledown)$, we will write $\mathsf{C}^{\bigtriangledown}$ for C, $\mathsf{T}^{\bigtriangledown}$ for $\mathsf{T}(\mathsf{C})$, $\Sigma^{\bigtriangledown}$ for $\Sigma(\mathsf{C}, \bigtriangledown)$. Moreover $A \leq_{\bigtriangledown} B$ will be short for $(A \leq B) \in \Sigma^{\bigtriangledown}$. Finally we will write $A \sim_{\bigtriangledown} B \iff A \leq_{\bigtriangledown} B \leq_{\bigtriangledown} A$.

In Figure 1 appears a list of special purpose axioms and rules which have been considered in the literature (the list of course is not exhaustive).

Notice that, when Ω is in $\mathsf{C}^{\bigtriangledown}$, the role of ν could be played also by the type $\Omega \to \Omega$. For this reason it is of no use to have at the same time in the language both ν and Ω. Hence we impose that the two constants do not occur together in any $\mathsf{C}^{\bigtriangledown}$. The elements Ω and ν play very special roles in the development of the theory. Therefore we stipulate the following blanket assumptions:

Assumption 1 : if $\Omega \in \mathsf{C}^{\bigtriangledown}$ then $(\Omega) \in \bigtriangledown$.
Assumption 2 : if $\nu \in \mathsf{C}^{\bigtriangledown}$ then $(\nu) \in \bigtriangledown$.

We introduce in Figure 2 a list of significant intersection-type theories which have been extensively considered in the literature. The order is logical, rather than historical: [3, 11, 2, 5]. We shall denote such theories as $\Sigma^{\bigtriangledown}$ with various different names $\bigtriangledown$, picked for mnemonic reasons. For each such $\bigtriangledown$ we specify in Figure 2 the type theory $\Sigma^{\bigtriangledown} = \Sigma(\mathsf{C}, \bigtriangledown)$ by giving the set of constants $\mathsf{C}^{\bigtriangledown}$ and the set $\bigtriangledown$ of extra axioms and rules taken from Figure 1. Here C_∞ is an infinite set of fresh atoms, i.e. different from Ω, ν.

Before giving the crucial notion of *intersection-type assignment system*, we need to introduce some preliminary definitions and notations.

A $\triangledown$-*basis* is a set of statements of the shape $x{:}B$, where $B \in \mathsf{T}^\triangledown$, all whose variables are distinct. An *intersection-type assignment system* relative to $\Sigma^\triangledown$, denoted by $\lambda\cap^\triangledown$, is a formal system for deriving judgments of the form $\Gamma \vdash^\triangledown M : A$, where the *subject* M is an untyped λ-term, the *predicate* A is in $\mathsf{T}^\triangledown$, and Γ is a $\triangledown$-basis. We will write $x \in \Gamma$ as short for $\exists A.(x{:}A) \in \Gamma$, i.e. x occurs as the subject of an assertion in Γ.

Definition 4 (Basic Type Assignment System)
Let $\Sigma^\triangledown$ *be a type theory. The* basic type assignment system, *denoted by* $\lambda\cap^\triangledown_{\mathcal{B}}$, *is a formal system for deriving judgments of the shape* $\Gamma \vdash^\triangledown_{\mathcal{B}} M : A$. *Its rules are the following:*

$$(Ax)\quad \frac{x{:}A \in \Gamma}{\Gamma \vdash^\triangledown_{\mathcal{B}} x{:}A} \qquad\qquad (\to \mathrm{I})\quad \frac{\Gamma, x{:}A \vdash^\triangledown_{\mathcal{B}} M : B}{\Gamma \vdash^\triangledown_{\mathcal{B}} \lambda x.M : A \to B}$$

$$(\to \mathrm{E})\quad \frac{\Gamma \vdash^\triangledown_{\mathcal{B}} M : A \to B \quad \Gamma \vdash^\triangledown_{\mathcal{B}} N : A}{\Gamma \vdash^\triangledown_{\mathcal{B}} MN : B} \qquad\qquad (\cap \mathrm{I})\quad \frac{\Gamma \vdash^\triangledown_{\mathcal{B}} M : A \quad \Gamma \vdash^\triangledown_{\mathcal{B}} M : B}{\Gamma \vdash^\triangledown_{\mathcal{B}} M : A \cap B}$$

$$(\leq_\triangledown)\quad \frac{\Gamma \vdash^\triangledown_{\mathcal{B}} M : A \quad A \leq_\triangledown B}{\Gamma \vdash^\triangledown_{\mathcal{B}} M : B}$$

The Basic Type Assignment System can be extended with other rules according to the set of constants belonging to $\mathsf{C}^\triangledown$ and the corresponding axioms and rules in $\Sigma^\triangledown$.

If $\Omega \in \mathsf{C}^\triangledown$, in line with the intended set-theoretic interpretation of Ω as the universe, we extend the Basic Type Assignment System with a suitable axiom for Ω.

Definition 5 (Ω-type Assignment System)
Let $\Sigma^\triangledown$ *be a type theory with* $\Omega \in \mathsf{C}^\triangledown$. *The axioms and rules of the* Ω-type assignment system *(denoted* $\lambda\cap^\triangledown_\Omega$*), are those of the Basic type Assignment System, together with the further axiom*

$(Ax\text{-}\Omega)\quad \Gamma \vdash^\triangledown_\Omega M : \Omega.$

Similarly, if $\nu \in \mathsf{C}^\triangledown$, in line with the intended interpretation of ν as the universe of *abstractions*, we define:

Definition 6 (ν-type Assignment System)
Let $\Sigma^\triangledown$ *be a type theory with* $\nu \in \mathsf{C}^\triangledown$. *The axioms and rules of the* ν-type assignment system *(denoted* $\lambda\cap^\triangledown_\nu$*), are those of the Basic Type Assignment System, together with the further axiom*

$(Ax\text{-}\nu)\quad \Gamma \vdash^\triangledown_\nu \lambda x.M : \nu.$

In the following $\lambda\cap^\triangledown$ will range over $\lambda\cap^\triangledown_{\mathcal{B}}$, $\lambda\cap^\triangledown_\Omega$ and $\lambda\cap^\triangledown_\nu$. More precisely we assume that $\lambda\cap^\triangledown$ stands for $\lambda\cap^\triangledown_\Omega$ whenever $\Omega \in \mathsf{C}^\triangledown$, for $\lambda\cap^\triangledown_\nu$ whenever $\nu \in \mathsf{C}^\triangledown$, and for $\lambda\cap^\triangledown_{\mathcal{B}}$ otherwise. Similarly for $\vdash^\triangledown$.

3 Applicative structures suitable for λ-calculus

In this section we introduce the semantical structures which we will consider in our investigation of soundness and completeness of intersection-type assignment systems.

Definition 7 (λ-applicative structures) *A* λ-applicative structure *is a triple* $\langle \mathcal{D}, \cdot, [\![\ ]\!]^{\mathcal{D}} \rangle$ *such that:*

1. $\langle \mathcal{D}, \cdot \rangle$ *is an applicative structure;*
2. $[\![\ ]\!]^{\mathcal{D}} : \Lambda \times \mathsf{Env}_{\mathcal{D}} \to \mathcal{D}$*, where* $\mathsf{Env}_{\mathcal{D}} = [Var \to \mathcal{D}]$*, is an interpretation function for λ-terms which satisfies:*
   (a) $[\![MN]\!]^{\mathcal{D}}_{\rho} = [\![M]\!]^{\mathcal{D}}_{\rho} \cdot [\![N]\!]^{\mathcal{D}}_{\rho}$;
   (b) $[\![\lambda x.M]\!]^{\mathcal{D}}_{\rho} = [\![\lambda y.M[x := y]]\!]^{\mathcal{D}}_{\rho}$ *if* $y \notin FV(M)$.

Notice that we do not postulate, in general, that $[\![x]\!]^{\mathcal{D}}_{\rho} = \rho(x)$. Nevertheless the class of environments which have this property will be of particular significance. Hence we put

Definition 8 (Good environments) *Let* $\langle \mathcal{D}, \cdot, [\![\ ]\!]^{\mathcal{D}} \rangle$ *be a λ-applicative structure. The environment* $\rho : Var \to \mathcal{D}$ *is* good *if for all* $x \in Var$*, we have* $[\![x]\!]^{\mathcal{D}}_{\rho} = \rho(x)$.

One can easily see that Plotkin's λ-structures as defined in [20], are λ-applicative structures in which all environments are good. In the next section we will introduce filter structures, which are again λ-applicative structures.

Models of, possibly restricted, λ-calculi as we commonly know them, can be viewed as special λ-applicative structures.

First we need to give the definition of *restricted* λ-calculus.

Definition 9 (Restricted λ-calculus) *Let* $R \subseteq \{\langle (\lambda x.M)N, M[x := N] \rangle \mid M, N \in \Lambda\}$. *The* restricted λ-calculus λ_R *is the calculus obtained from the standard λ-calculus, by restricting the* β *rule to the redexes in R.*

Clearly when $R = \beta$, λ_R is the standard λ-calculus. The main examples of *truly* restricted λ-calculi are Plotkin's call-by-value λ_v-calculus and the λ-**I**-**N**-calculus of [16]. Finally we give the crucial definition

Definition 10 ((Restricted) λ-model) *A* (restricted) λ-model *for the (restricted) λ-calculus* λ_R*, is a λ- applicative structure,* $\langle \mathcal{D}, \cdot, [\![\ ]\!]^{\mathcal{D}} \rangle$*, which moreover satisfies*

$$[\![(\lambda x.M)N]\!]^{\mathcal{D}}_{\rho} = [\![M[x := N]]\!]^{\mathcal{D}}_{\rho} \text{ for } \langle (\lambda x.M)N, M[x := N] \rangle \in R.$$

It is easy to see that all notions of models for, possibly restricted, λ-calculi, based on applicative structures, can be cast in the above setting.

4 Filter structures and interpretation of λ-terms

In this section we introduce *filter structures*. These are the basic tools for building λ-applicative structures, in effect λ-models, which realize completeness for intersection-type theories.

Filter structures arise naturally in the context of those generalizations of Stone duality that are used in discussing domain theory in logical form (see [1], [9], [26]). This approach provides a conceptually independent semantics to intersection-types, the *lattice semantics*. Types are viewed as *compact elements* of domains. The type Ω denotes the least element, intersections denote joins of compact elements, and arrow types allow to internalize the space of continuous endomorphisms. Following the paradigm of Stone duality, type theories give rise to *filter structures*, where the interpretation of λ-terms can be given through a finitary logical description.

We start by introducing the notion of *filter* of types. Then we show how to associate to each type theory its filter structure. This is a λ-applicative structure where the interpretation of a λ-term is given by the filter of the types which can be assigned to it.

Definition 11 *Let $\Sigma^\triangledown$ be a type theory.*

1. *A $\triangledown$-filter (or a filter over $\mathsf{T}^\triangledown$) is a set $X \subseteq \mathsf{T}^\triangledown$ such that*
   (a) *if $\Omega \in \mathsf{C}^\triangledown$ then $\Omega \in X$;*
   (b) *if $A \leq_\triangledown B$ and $A \in X$, then $B \in X$;*
   (c) *if $A, B \in X$, then $A \cap B \in X$;*
2. *$\mathcal{F}^\triangledown$ denotes the set of $\triangledown$-filters over $\mathsf{T}^\triangledown$;*
3. *if $X \subseteq \mathsf{T}^\triangledown$, $\uparrow X$ denotes the $\triangledown$-filter generated by X;*
4. *a $\triangledown$-filter is* principal *if it is of the shape $\uparrow \{A\}$, for some type A. We shall denote $\uparrow \{A\}$ simply by $\uparrow A$.*

Notice that $\uparrow \emptyset$ is the $\triangledown$-filter $\uparrow \Omega$, if $\Omega \in \mathsf{C}^\triangledown$, and $\emptyset$ otherwise.

The next step is to define application over sets of filters.

Definition 12 (Filter structure) *Let $\Sigma^\triangledown$ be a type theory.*

1. *Application $\cdot : \mathcal{F}^\triangledown \times \mathcal{F}^\triangledown \to \mathcal{F}^\triangledown$ is defined as*
$$X \cdot Y = \uparrow \{B \mid \exists A \in Y.\, A \to B \in X\}.$$
2. *The maps $F^\triangledown : \mathcal{F}^\triangledown \to [\mathcal{F}^\triangledown \to \mathcal{F}^\triangledown]$ and $G^\triangledown : [\mathcal{F}^\triangledown \to \mathcal{F}^\triangledown] \to \mathcal{F}^\triangledown$ are defined by:*
$$\begin{array}{rcl} F^\triangledown(X) & = & \lambda\!\!\lambda Y \in \mathcal{F}^\triangledown.\, X \cdot Y; \\ G^\triangledown(f) & = & \begin{cases} \uparrow \{A \to B \mid B \in f(\uparrow A)\} \cup \uparrow \nu & \text{if } \nu \in \mathsf{C}^\triangledown \\ \uparrow \{A \to B \mid B \in f(\uparrow A)\} & \text{otherwise.} \end{cases} \end{array}$$

   The triple $\langle \mathcal{F}^\triangledown, F^\triangledown, G^\triangledown \rangle$ is called the filter structure *over $\Sigma^\triangledown$.*

Filter structures induce immediately λ-applicative structures.

Proposition 1 *Let* $\langle \mathcal{F}^{\nabla}, F^{\nabla}, G^{\nabla}\rangle$ *be a filter structure. Let* ρ *range over the set of term environments* $\mathsf{Env}_{\mathcal{F}^{\nabla}} = [Var \to \mathcal{F}^{\nabla}]$*. Define the interpretation function* $[\![\ ]\!]^{\mathcal{F}^{\nabla}} : \Lambda \times \mathsf{Env}^{\nabla}_{\mathcal{F}} \to \mathcal{F}^{\nabla}$ *as follows:*

- *if there exists* $x \in Var$ *such that* $\rho(x) = \emptyset$*, then* $[\![M]\!]^{\mathcal{F}^{\nabla}}_{\rho} = \emptyset$*;*
- *otherwise put inductively:*

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

The triple $\langle \mathcal{F}^{\nabla}, \cdot, [\![\ ]\!]^{\mathcal{F}^{\nabla}}\rangle$ *is a λ-applicative structure .*

The interpretation function of a term coincides with the set of types which are derivable for it. This will be a crucial property in showing completeness using filter structures.

Theorem 1 *For any λ-term M and environment* $\rho : Var \to \mathcal{F}^{\nabla}$*,*

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

where $\Gamma \models \rho$ *if and only if for all* $x \in Var$*,* $\rho(x) \neq \emptyset$*, and moreover* $(x : B) \in \Gamma$ *implies* $B \in \rho(x)$.

5 Set-theoretic semantics of intersection-types

This is the main section of the paper. Here, we discuss completeness for the three *set-theoretic semantics* of intersection-types mentioned in the introduction. In particular, we characterize those intersection-type theories which induce complete type assignment systems for the *inference*, the *simple* and the *F*- semantics, over λ-applicative structures. As we will see these conditions apply also to the preorders which induce complete systems with respect to the three semantics over λ-models. We recall that according to these semantics the meaning of types are *subsets* of the universe of discourse, i.e. the applicative structure. The "intersection" type constructor is always interpreted as the *set-theoretic intersection*. While, the "arrow" is interpreted as a *logical predicate*, i.e. the set of those points, which belong to a suitable distinguished set, and whose applicative behavior is that of mapping the antecedent of the arrow into the consequent.

As we remarked earlier the very existence of complete type assignment systems for such semantics over λ-applicative structures, is one of the strongest motivations for the whole enterprise of developing a theory of intersection-types.

In discussing completeness, *soundness* is not really an issue, since all intersection-type theories are sound. To achieve *adequacy* and hence *completeness* we have to restrict to two *disjoint* classes of type theories, namely the *natural* theories and the *strict* theories. Filter structures are essential to showing adequacy. In such structures, in fact, the set-theoretic interpretation of a type, as an appropriate subset, is in one-to-one correspondence with the *principal* filter generated by that type.

We write "the type theory $\Sigma^{\triangledown}$ validates $\triangledown'$" to mean that all axioms and rules of $\triangledown'$ are admissible in $\Sigma^{\triangledown}$.

Definition 13 1. *A type theory* $\Sigma^{\triangledown}$ *is* strict *if* $\Omega \notin \mathsf{C}^{\triangledown}$ *and it validates* $\mathcal{Ba}$ *as defined in Figure 2.*

2. *A type theory* $\Sigma^{\triangledown}$ *is* natural *if* $\Omega \in \mathsf{C}^{\triangledown}$ *and it validates* $\mathcal{AO}$ *as defined in Figure 2.*

Notice that by the blanket assumption that axiom $(\nu) \in \triangledown$ whenever $\nu \in \mathsf{C}^{\triangledown}$ (cf. page 4), a strict type theory containing the constant ν validates $\mathcal{EHR}$. All the theories appearing in Figure 2 are natural, if they contain Ω, and strict otherwise.

Type interpretations can be given on λ-applicative structures once we have fixed a distinguished set Φ of functional objects. There are various choices for this set. Amongst these there is a *maximal one* and a *minimal* one. The former determines the *simple* semantics, the latter the *F-semantics*. Admittedly our notion of *F-semantics* could be slightly generalized in the case of λ-applicative structures, which are not λ*-models*.

Definition 14 (Type Interpretation Domain)

1. *A* type interpretation domain *is a quadruple* $\mathcal{I} = \langle \mathcal{D}, \cdot, [\![\ ]\!]^{\mathcal{D}}, \Phi \rangle$ *such that:*
   - $\langle \mathcal{D}, \cdot, [\![\ ]\!]^{\mathcal{D}} \rangle$ *is a* λ*-applicative structure;*
   - Φ *is a subset of* $\mathcal{D}$*, called the* functionality set*, such that* $[\![\lambda x.M]\!]^{\mathcal{D}}_{\rho} \in \Phi$ *for all* x, M, ρ.
2. *A type interpretation domain* $\mathcal{I} = \langle \mathcal{D}, \cdot, [\![\ ]\!]^{\mathcal{D}}, \Phi \rangle$ *is a* simple interpretation domain *if* $\Phi = \mathcal{D}$*;*
3. *A type interpretation domain* $\mathcal{I} = \langle \mathcal{D}, \cdot, [\![\ ]\!]^{\mathcal{D}}, \Phi \rangle$ *is an* F-interpretation domain *if* $\Phi = \{d \in \mathcal{D} \mid d = [\![\lambda x.M]\!]^{\mathcal{D}}_{\rho}$ *for some* $x, M, \rho\}$.

Definition 15 (Type Interpretation) *Let* $\mathcal{I} = \langle \mathcal{D}, \cdot, [\![\ ]\!]^{\mathcal{D}}, \Phi \rangle$ *be a type interpretation domain. The* type interpretation *induced by the type environment* $\mathcal{V} : \mathsf{C}^{\triangledown} \to \mathsf{P}(\mathcal{D})$ *is defined by:*

1. $[\![\Omega]\!]^{\mathcal{I}}_{\mathcal{V}} = \mathcal{D}$ *;*
2. $[\![\nu]\!]^{\mathcal{I}}_{\mathcal{V}} = \Phi$*;*
3. $[\![\psi]\!]^{\mathcal{I}}_{\mathcal{V}} = \mathcal{V}(\psi)$ *if* $\psi \in \mathsf{C}^{\triangledown}$ *and* $\psi \neq \Omega, \nu$*;*
4. $[\![A \to B]\!]^{\mathcal{I}}_{\mathcal{V}} = \{d \in \Phi \mid \forall e \in [\![A]\!]^{\mathcal{I}}_{\mathcal{V}}.\ d \cdot e \in [\![B]\!]^{\mathcal{I}}_{\mathcal{V}}\}$*;*
5. $[\![A \cap B]\!]^{\mathcal{I}}_{\mathcal{V}} = [\![A]\!]^{\mathcal{I}}_{\mathcal{V}} \cap [\![B]\!]^{\mathcal{I}}_{\mathcal{V}}$.

This definition is the counterpart for intersection-types of the *inference semantics* for polymorphic types of [19], generalized by allowing $\langle \mathcal{D}, \cdot, [\![\,]\!]^{\mathcal{D}} \rangle$ to be just a λ-applicative structure instead of a λ-model.

In discussing *sound* type assignment systems we consider only type interpretation domains and type environments which are good and which agree with the inclusion relation between types in the following sense:

Definition 16 *A type interpretation domain* $I = \langle \mathcal{D}, \cdot, [\![\,]\!]^{\mathcal{D}}, \Phi \rangle$ *and a type environment* $\mathcal{V} : \mathsf{C}^{\nabla} \to \mathsf{P}(\mathcal{D})$

- *are* good *if for all* $A, B \in \mathsf{T}^{\nabla}$
  1. *for all* good *environments* ρ *and* $d \in [\![A]\!]^{I}_{\mathcal{V}}$, $\rho[x := d]$ *is* good*;*
  2. *for all* good *environments* ρ*, terms* M *and variables* x*,*
     $\forall d \in [\![A]\!]^{I}_{\mathcal{V}}.\ [\![M]\!]^{\mathcal{D}}_{\rho[x:=d]} \in [\![B]\!]^{I}_{\mathcal{V}} \Rightarrow \forall d \in [\![A]\!]^{I}_{\mathcal{V}}.[\![\lambda x.M]\!]^{\mathcal{D}}_{\rho} \cdot d \in [\![B]\!]^{I}_{\mathcal{V}}$
- agree *with a type theory* Σ^{∇} *iff for all* $A, B \in \mathsf{T}^{\nabla}$

$$A \leq_{\nabla} B \ \Rightarrow\ [\![A]\!]^{I}_{\mathcal{V}} \subseteq [\![B]\!]^{I}_{\mathcal{V}}.$$

For lack of space we cannot discuss conditions which imply goodness and agreement. We only remark that they hold whenever we consider models of restricted λ-calculi.

We now introduce formally the three semantics. The definitions are natural given how we have argued informally so far, but for the restriction (in the definition of $\models^{\nabla}$) to those type interpretation domains and type environments which are good and agree with Σ^{∇}.

Definition 17 (Semantic Satisfiability) *Let* $I = \langle \mathcal{D}, \cdot, [\![\,]\!]^{\mathcal{D}}, \Phi \rangle$ *be a type interpretation domain.*

1. $I, \rho, \mathcal{V} \models M : A$ *iff* $[\![M]\!]^{\mathcal{D}}_{\rho} \in [\![A]\!]^{I}_{\mathcal{V}}$*;*
2. $I, \rho, \mathcal{V} \models \Gamma$ *iff* $I, \rho, \mathcal{V} \models x : B$ *for all* $(x{:}B) \in \Gamma$*;*
3. $\Gamma \models^{\nabla}_{i} M : A$ *iff* $I, \rho, \mathcal{V} \models \Gamma$ *implies* $I, \rho, \mathcal{V} \models M : A$ *for all good type interpretation domains* I *and type environments* $\mathcal{V}$ *which moreover agree with* Σ^{∇}*, and for all good term environments* ρ*;*
4. $\Gamma \models^{\nabla}_{s} M : A$ *iff* $I, \rho, \mathcal{V} \models \Gamma$ *implies* $I, \rho, \mathcal{V} \models M : A$ *for all good simple interpretations domains* I *and type environments* $\mathcal{V}$ *which moreover agree with* Σ^{∇}*, and for all good term environments* ρ*;*
5. $\Gamma \models^{\nabla}_{F} M : A$ *iff* $I, \rho, \mathcal{V} \models \Gamma$ *implies* $I, \rho, \mathcal{V} \models M : A$ *for all good F-interpretations domains* I*, and type environments* $\mathcal{V}$ *which moreover agree with* Σ^{∇}*, and for all good term environments* ρ.

In view of the above definition, we can say that the *inference semantics* is given by $\models^{\nabla}_{i}$, the *simple semantics* by $\models^{\nabla}_{s}$, and the *F-semantics* by $\models^{\nabla}_{F}$.

We shall denote with $\models^{\nabla}$ any of the three $\models^{\nabla}_{i}$, $\models^{\nabla}_{s}$, and $\models^{\nabla}_{F}$.

Derivability in the type system implies semantic satisfiability, as shown in the next theorem. Its proof by induction on derivations is straightforward.

Theorem 2 (Soundness) *$\Gamma \vdash^{\nabla} M : A$ implies $\Gamma \models^{\nabla} M : A$.*

As far as adequacy, first we observe that only natural or strict type theories can be adequate.

Proposition 2 *(Adequacy implies naturality or strictness) If $\Gamma \models^{\nabla} M : A$ implies $\Gamma \vdash^{\nabla} M : A$ for all Γ, M, A, then Σ^{∇} is a natural or a strict type theory.*

Now we shall discuss adequacy for each of the three semantics separately.

First we consider the inference semantics. Our result is that all natural and all strict type theories are adequate for the inference semantics.

Theorem 3 *(Naturality or strictness imply adequacy) Let Σ^{∇} be a natural or a strict type theory, then $\Gamma \models_i^{\nabla} M : A$ implies $\Gamma \vdash^{\nabla} M : A$.*

Hence the only type theories which turn out to be complete with respect to the inference semantics are the natural and the strict type theories. There are of course many theories of interest which do not belong to these classes. For instance the intersection-type theory which induces the filter structure isomorphic to Scott's $\mathcal{P}(\omega)$ is such a theory. The reader can see [4] for more examples.

Notice that the theory $\Sigma^{\mathcal{BCD}}$ induces a filter structure which is a λ-model [5], the theory $\Sigma^{\mathcal{EHR}}$ induces a model for the λ_v-calculus [11], and the theory $\Sigma^{\mathcal{B}a}$ induces a model for the λ-**I**-**N**-calculus [16]. Hence we have that natural theories, which induce λ-models, are complete also for the class of λ-models. And moreover, strict theories, which induce models of the other two restricted λ-calculi, are complete for the corresponding classes of models.

Now we characterize those theories which are complete with respect to the simple semantics.

Theorem 4 *(Adequacy for the simple semantics) $\Gamma \models_s^{\nabla} M : A$ implies $\Gamma \vdash^{\nabla} M : A$ iff Σ^{∇} is a strict type theory or a natural type theory which validates axiom (Ω-η).*

Among the type theories of Figure 2, those adequate for the simple semantics are $\Sigma^{\mathcal{B}a}$ and $\Sigma^{\mathcal{BCD}}$. Other adequate type theories in the literature are those of [16, 11, 21, 8, 17]. Non adequate type theories are all those inducing computationally adequate models for the lazy λ-calculus, e.g. $\Sigma^{\mathcal{AO}}$. The same argument used for the inference semantics allows to show that the natural theories mentioned in Theorem 4, which induce λ-models, are precisely those which are complete also for the class of λ-models. Moreover the strict theories, which induce λ_v-models and λ-**I**-**N**-models, are complete for the corresponding classes of models. The completeness for the simple semantics of $\lambda\cap_{\Omega}^{\nabla}$ whenever Σ^{∇} validates $\mathcal{BCD}$ and $\langle \mathcal{F}^{\nabla}, \cdot, [\![\;]\!]_{\nabla} \rangle$ is a λ-model was proved in [9] using a filter model and in [8] using the term model of β-equality.

Finally we turn to the F-semantics. A first observation is that an F-interpretation $I = \langle \mathcal{D}, \cdot, [\![\;]\!]^{\mathcal{D}}, \Phi \rangle$ and a type environment $\mathcal{V}$ do agree with the natural type theories which validate axiom (Ω-η) only when $\Phi = \mathcal{D}$. In fact we have $[\![\Omega]\!]^{\mathcal{D}}_{\mathcal{V}} = \mathcal{D}$ and $[\![\Omega \to \Omega]\!]^{\mathcal{D}}_{\mathcal{V}} = \{d \in \Phi \mid d = [\![\lambda x.M]\!]^{\mathcal{D}}_{\rho}$ for some $x, M, \rho\} = \Phi$. So for these type theories the F-semantics coincides with the simple semantics. Therefore it is interesting to consider only strict type theories or natural type theories such that $\Omega \not\sim_{\nabla} \Omega \to \Omega$.

The following definition singles out the type theories which do not validate axiom (Ω-η) but which are adequate for the F-semantics as stated in Theorem 5.

Definition 18 *A type theory Σ^{∇} is an* F-type theory *iff*

1. *either Σ^{∇} is a strict type theory such that $\nu \notin \mathsf{C}^{\nabla}$ and for all $\psi \in \mathsf{C}^{\nabla}$, $A, B \in \mathsf{T}^{\nabla}$, there are $I, A_i, B_i \in \mathsf{T}^{\nabla}$ such that $\psi \cap (A \to B) \sim_{\nabla} \cap_{i \in I} (A_i \to B_i)$;*
2. *or Σ^{∇} is a strict type theory such that $\nu \in \mathsf{C}^{\nabla}$ and for all $\psi \in \mathsf{C}^{\nabla}$ either $\nu \leq_{\nabla} \psi$ or there are $I, A_i, B_i \in \mathsf{T}^{\nabla}$ such that $\psi \cap \nu \sim_{\nabla} \cap_{i \in I} (A_i \to B_i)$;*
3. *or Σ^{∇} is a natural type theory such that $\Omega \not\sim_{\nabla} \Omega \to \Omega$ and for all $\psi \in \mathsf{C}^{\nabla}$ there are $I, A_i, B_i \in \mathsf{T}^{\nabla}$ such that $\psi \cap (\Omega \to \Omega) \sim_{\nabla} \cap_{i \in I} (A_i \to B_i)$.*

For example $\Sigma^{\mathcal{EHR}}$ and $\Sigma^{\mathcal{AO}}$ are F-type theories.

Notice that, if $\nu \in \mathsf{C}^{\nabla}$, condition (2) of the above definition implies that for all $\psi \in \mathsf{C}^{\nabla}$, $A, B \in \mathsf{T}^{\nabla}$, there are $I, A_i, B_i \in \mathsf{T}^{\nabla}$ such that $\psi \cap (A \to B) \sim_{\nabla} \cap_{i \in I} (A_i \to B_i)$. Similarly, if $\Omega \in \mathsf{C}^{\nabla}$, condition (3) of the above definition implies that for all $\psi \in \mathsf{C}^{\nabla}$, $A, B \in \mathsf{T}^{\nabla}$, there are $I, A_i, B_i \in \mathsf{T}^{\nabla}$ such that $\psi \cap (A \to B) \sim_{\nabla} \cap_{i \in I} (A_i \to B_i)$.

Theorem 5 *(Adequacy for the F-semantics) $\Gamma \models^{\nabla}_{F} M : A$ implies $\Gamma \vdash^{\nabla} M : A$ iff either Σ^{∇} is an F-type theory or Σ^{∇} is a natural type theory which validates axiom (Ω-η).*

The theories $\Sigma^{\mathcal{EHR}}$, $\Sigma^{\mathcal{AO}}$ and $\Sigma^{\mathcal{BCD}}$, as well as the type theories of [16, 21, 8, 17] are adequate for the F-semantics. Moreover for the last five and $\Sigma^{\mathcal{BCD}}$ the simple semantics coincides with the F-semantics. The theory $\Sigma^{\mathcal{B}a}$ is an example of a theory which is not adequate with respect to the F-semantics. The remark concerning λ-models and restricted λ-models made for the inference and the simple semantics, applies also to the F-semantics.

6 Related work and final remarks

In the literature there are essentially five ways of interpreting Curry's types in a model of the untyped λ-calculus. They differ in the interpretation of the *arrow* type constructor. In what follows we shall mainly follow the terminology of [14].

The simple and the F-semantics are defined as expected.

Following [24], the *quotient set semantics* takes into account that we want to consider equivalent two functions iff they give equivalent results when applied to equivalent arguments. So types are interpreted as partial equivalence relations of the domain rather than simply as subsets. The arrow constructor is defined as for logical relations: $d \sim_{A \to B} d'$ iff for all c, c' such that $c \sim_A c'$ it holds $d \cdot c \sim_B d' \cdot c'$.

The *F-quotient set semantics* [23], modifies the quotient set semantics, in the same way as the F-semantics modifies the simple semantics. Namely it requires that all elements of the domain which are equivalent with respect to an arrow must be canonical representatives of functions.

Finally, Mitchell in [19] introduces another semantics, which he calls *inference semantics*, in which the interpretation of the arrow must *at least* contain the canonical representatives of functions which behave correctly with respect to the application.

All the above semantics easily extend to *intersection-types* [5, 10] and to *polymorphic types* [19].

The crucial question in the semantics of types is the completeness of type assignment systems. Hindley proved in [14] that Curry's type assignment system is complete for all the mentioned semantics. More specifically [14] and [15] show the completeness for the simple semantics and moreover that:

1. $\Gamma \models_F^{\bigtriangledown} M : A$ if and only if $\Gamma \models_s^{\bigtriangledown} M : A$, when A is a Curry type;
2. the simple semantics is a special case of the quotient set semantics;
3. the F-semantics is a special case of the F-quotient set semantics.

The argument showing points (2) and (3) easily extends to intersection and polymorphic types, so for these type disciplines it is enough to discuss only completeness for the simple and the F-semantics to get completeness results for the quotiented versions. One could define also a quotiented version of the inference semantics, but this would be treated similarly.

Completeness of various intersection-type assignment systems, over various kinds of λ-models, have been proved in [5, 13, 9, 8, 3, 2, 18].

As far as the completeness with respect to the F-semantics of intersection-type assignment systems, over λ-models, we can cite [10, 27, 2]. In [10] the intersection-type assignment systems which are complete for the F-semantics are characterized. More specifically it is shown that an intersection-type assignment system is complete for the F-semantics if and only if types are invariant under β-equality of subjects and moreover the following rule (due to R.Hindley):

$$\text{(Hindley rule)} \quad \frac{\Gamma \vdash^{\bigtriangledown} M : A \cap (\Omega^n \to \Omega) \;\; x_i \notin FV(M) \; (1 \le i \le n)}{\Gamma \vdash^{\bigtriangledown} \lambda x_1 \ldots x_n . M x_1 \ldots x_n : A}$$

where A is an arbitrary type and Ω is the universal type, is a derived rule.

Yokouchi in [27] shows that if we add two suitable rules (quite similar to Hindley rule) to the intersection-type assignment system of [7] we obtain completeness for the F-semantics.

Abramsky and Ong in [2] prove the completeness of the theory $\Sigma^{\mathcal{AO}}$, with respect to the F-semantics, over applicative structures with convergence.

We conclude the paper with two final remarks.

If $\Sigma^\bigtriangledown$ is a natural type theory which is adequate for the F-semantics, then Hindley's rule is admissible in $\lambda\cap^\bigtriangledown_\Omega$. This follows from the observation that under the given conditions for all $n \geq 0$ and for all $\psi \in \mathbb{C}^\bigtriangledown$ we can find $I, A_i^{(1)}, \ldots, A_i^{(n)}, B_i$ such that

$$\psi \cap (\Omega^n \to \Omega) \sim_\bigtriangledown \cap_{i\in I} (A_i^{(1)} \to \ldots \to A_i^{(n)} \to B_i).$$

For lack of space-time, in this paper we have not discussed Kripke-like semantics, a full treatment will appear in [4].

Acknowledgments The authors are grateful to Yoko Motohama for her helpful comments.

References

[1] S. Abramsky [1991]. Domain theory in logical form, *Ann. Pure Appl. Logic* **51**(1-2), pp. 1–77.

[2] S. Abramsky and C.-H.L. Ong [1993]. Full abstraction in the lazy lambda calculus, *Inform. Comput.* **105**(2), pp. 159–267.

[3] S. van Bakel [1992]. Complete restrictions of the intersection type discipline, *Theoret. Comput. Sci.* **102**(1), pp. 135–163.

[4] H. Barendregt *et al* [2000]. *Typed λ-calculus and applications*, North Holland (to appear).

[5] H. Barendregt, M. Coppo and M. Dezani-Ciancaglini [1983]. A filter lambda model and the completeness of type assignment, *J. Symbolic Logic* **48**(4), pp. 931–940.

[6] M. Coppo and M. Dezani-Ciancaglini [1980]. An extension of the basic functionality theory for the λ-calculus, *Notre Dame J. Formal Logic* **21**(4), pp. 685–693.

[7] M. Coppo, M. Dezani-Ciancaglini and B. Venneri [1981]. Functional characters of solvable terms, *Z. Math. Logik Grundlag. Math.* **27**(1), pp. 45–58.

[8] M. Coppo, M. Dezani-Ciancaglini and M. Zacchi [1987]. Type theories, normal forms, and D_∞-lambda-models, *Inform. Comput.* **72**(2), pp. 85–116.

[9] M. Coppo, F. Honsell, M. Dezani-Ciancaglini and G. Longo [1984]. Extended type structures and filter lambda models, *Logic Colloquium '82* , North-Holland, Amsterdam, pp. 241–262.

[10] M. Dezani-Ciancaglini and I. Margaria [1986]. A characterization of F-complete type assignments, *Theoret. Comput. Sci.* **45**(2), pp. 121–157.

[11] L. Egidi, F. Honsell and S. Ronchi Della Rocca [1992]. Operational, denotational and logical descriptions: a case study, *Fund. Inform.* **16**(2), pp. 149–169.

[12] E. Engeler [1981]. Algebras and combinators, *Algebra Universalis* **13**(3), pp. 389–392.

[13] J.R. Hindley, [1982]. The simple semantics for Coppo–Dezani–Sallé types, *ISP'82*, LNCS 137, Springer, Berlin, pp. 212–226.

[14] J.R. Hindley, [1983]. The completeness theorem for typing λ-terms, *Theoret. Comput. Sci.* **22**, pp. 1–17.

[15] J.R. Hindley, [1983]. Curry's types are complete with respect to the F-semantics too, *Theoret. Comput. Sci.* **22**, pp. 127–133.

[16] F. Honsell and M. Lenisa [1999]. Semantical analysis of perpetual strategies in λ-calculus, *Theoret. Comput. Sci.* **212**(1-2), pp. 183–209.

[17] F. Honsell and S. Ronchi Della Rocca [1992]. An approximation theorem for topological lambda models and the topological incompleteness of lambda calculus, *J. Comput. System Sci.* **45**(1), pp. 49–75.

[18] M.Lenisa, [1997]. Semantic techniques for deriving coinductive characterizations of observational equivalences for λ-calculi, *TLCA'97*, LNCS 1210, Springer, Berlin, pp. 248–266.

[19] J.Mitchell [1988]. Polymorphic type inference and containment, *Inform. Comput.* **76**, pp. 211–249.

[20] G.D. Plotkin [1993]. Set-theoretical and other elementary models of the λ-calculus, *Theoret. Comput. Sci.* **121**(1-2), pp. 351–409.

[21] D.S. Scott [1972]. Continuous lattices, *Toposes, algebraic geometry and logic* , Springer, Berlin, pp. 97–136. LNM, Vol. 274.

[22] D.S. Scott [1975]. Open problem, *Lambda Calculus and Computer Science Theory*, LNCS 37, Springer, Berlin, p. 369.

[23] D.S. Scott [1976]. Data types as lattices, *SIAM J. Comput.* **5**(3), pp. 522–587.

[24] D.S. Scott [1980]. Lambda Calculus: some models, some philosophy, *The Kleene Symposium*, North-Holland, Amsterdam, pp. 223–266.

[25] D.S. Scott [1980]. Letter to Albert Meyer.

[26] S. Vickers [1989]. *Topology via logic*, Cambridge University Press, Cambridge.

[27] H. Yokouchi [1994]. F-semantics for type assignment systems, *Theoret. Comput. Sci.* **129**, pp. 39-77.

Fabio Alessi is with the Dipartimento di Matematica ed Informatica, via delle Scienze 208, 33100 Udine, Italy. E-mail: alessi@dimi.uniud.it

Mariangiola Dezani-Ciancaglini is with the Dipartimento di Informatica, corso Svizzera 185, 10149 Torino, Italy. E-mail: dezani@di.unito.it

Furio Honsell is with the Dipartimento di Matematica ed Informatica, via delle Scienze 208, 33100 Udine, Italy. E-mail: honsell@dimi.uniud.it