A Note on Logic Programming Fixed-Point Semantics

Vladimiro Sassone

Dipartimento di Informatica – Università di Pisa
e-mail: `vladi@di.unipi.it`

Abstract

In this paper, we present an account of classical Logic Programming fixed-point semantics in terms of two standard categorical constructions in which the least Herbrand model is characterized by properties of universality.

In particular, we show that, given a program P, the category of models of P is reflective in the category of interpretations for P. In addition, we show that the immediate consequence operator gives rise to an endofunctor $\mathcal{T}_P$ on the category of Herbrand interpretations for P such that category of algebras for $\mathcal{T}_P$ is the category of Herbrand models of P.

As consequences, we have that the least Herbrand model of P is the least fixed-point of $\mathcal{T}_P$ and is the reflection of the empty Herbrand interpretation.

Introduction

Logic Programming, arisen in the early seventies from the work on automatic theorem proving, is a very simple formalism based on the rigorous mathematical framework of first-order theories. The revolutionary idea introduced in Kowalski's work [13] is that logic has a *computational interpretation* and, therefore, logic can be used as a programming language.

Generally speaking, following [17], a logic system is a formal system consisting of the set of sentences over a certain alphabet equipped with an *entailment relation*—informally speaking a proof system capable to "calculate" consequences of sets of sentences—and a *satisfaction relation* related by a condition of soundness, but not necessarily of completeness, of entailment with respect to satisfaction. From the Logic Programming point of view, any such system is a programming language whose operational semantics is given by the entailment relation.

The classical theory of Logic Programming is devoted to the study of the fragment of first-order logic consisting of *Horn clauses*. Although Horn clauses reduce the expressive power of first-order logic, the choice of such a fragment presents some advantages both from the computational and from the model theoretic point of view. In fact, on one hand it has a simple, efficient goal-oriented deduction system (*SLD-resolution*) while on the other hand it is the largest fragment of first-order logic such that every set of formulas (program) admits an *initial model*, i.e., a model in which the facts entailed by the program are true and everything else is false. Moreover, Horn clauses have a nice computational interpretation as function calls in functional programming.

The latter fact led to the development of a *functional* (or fixed-point) *semantics* for logic programs with which we are concerned in this note and that we recall in Section 1.

In recent years, *algebraic* ([10, 11]) and *categorical* ([2, 3, 8, 17]) methods have been applied in the study of logic programs. The work here lies between these approaches in the sense that we follow the algebraic style of Goguen and Meseguer and study how well-known categorical constructions relate to standard logic programs fixed-point semantics. Doing that, we invariantly find the *least Herbrand model* as a distinguished element in those constructions.

Among the most studied categorical constructions, there are *adjunctions* and *algebras for endofunctors* ([12, 15, 4, 5]).

Algebras for endofunctors are a translation to the categorical language and generalization to arbitrary categories of the classical notion of algebra built over a set of elements, where the role of the signature is played by functors. Equipped with a sensible notion of morphism, the algebras for a given endofunctor $\mathcal{T}$ form a category which provides the mechanism to define the concepts of fixed-point and least fixed-point for $\mathcal{T}$.

In Section 2.1, we show that, fixed a program P, the immediate consequence operator can be lifted to an endofunctor $\mathcal{T}_P$ on the category of Herbrand interpretations, in such a way that the category of algebras for such a functor is exactly the category of Herbrand models of P. Moreover, we show that the least Herbrand model of P is the least fixed-point of $\mathcal{T}_P$.

Universal and free constructions appear everywhere in Mathematics and Computer Science and the relevance of adjunctions follows exactly from the fact that they elegantly describe such situations. Particular forms of adjunctions are the *reflections*. Very informally, a reflection of a category $\mathbf{B}$ to a subcategory $\mathbf{A}$ guarantees the existence of a canonical representative in $\mathbf{A}$ for each object in $\mathbf{B}$. In Section 2.2, we show that the category of models of a program is *reflective* in the category of interpretations for that program and that the least Herbrand model is the *reflection* of the empty Herbrand interpretation.

1 Fixed-Point Semantics for Logic Programs

In this section, we briefly recall the basic definitions and results of fixed-point semantics for logic programs ([1, 14, 16], among the others) in the algebraic style of [10, 11].

1.1 Logic Programs

Syntactically, logic programs are terms of languages in which at least two different kinds of entities can be recognized: operators (or constructors) and predicates. Such a nature is faithfully taken into account by signatures with predicates.

Definition 1.1 *(Signatures with Predicates)*
A (one-sorted) signature with predicates *is a pair* $\langle \Sigma, \Pi \rangle$ *where* Σ *and* Π *are disjoint families of disjoint sets of, respectively, symbols for operations* $\{\Sigma_n \mid n \in \mathbb{N}\}$ *and symbols for predicates* $\{\Pi_n \mid n \in \mathbb{N}\}$. Σ_n *and* Π_n *contain the symbols of arity* n. ✓

Given a signature $\langle \Sigma, \Pi \rangle$ and a set of symbols for variables X, that without loss of generality we will suppose disjoint from any Σ_i, terms built up from constructors and variables represent individuals to whom predicates may be applied.

Definition 1.2 *(Terms)*
The set $T_{\Sigma,\Pi}(X)$ *of* terms *with variables* X *on the signature* $\langle\Sigma,\Pi\rangle$ *is the smallest set such that:*

(i) $\Sigma_0 \cup X \subseteq T_{\Sigma,\Pi}(X)$;

(ii) $\forall t_1,\ldots,t_n \in T_{\Sigma,\Pi}(X)$ *and* $\forall \sigma \in \Sigma_n$, $\sigma(t_1,\ldots,t_n) \in T_{\Sigma,\Pi}(X)$ ✓

Definition 1.3 *(Atoms)*
The set of atoms *with variables* X *on* (Σ,Π) *is the set of formulas*

$$B_{\Sigma,\Pi}(X) = \Big\{\rho(t_1,\ldots,t_n) \;\Big|\; \rho \in \Pi_n \text{ and } t_1,\ldots,t_n \in T_{\Sigma,\Pi}(X)\Big\}.$$

✓

Given $B \subseteq B_{\Sigma,\Pi}(X)$, we will denote by $[\![B]\!]_\rho$ the set $\Big\{(t_1,\ldots,t_n) \;\Big|\; \rho(t_1,\ldots,t_n) \in B\Big\}$.

When the set of variables is the empty, the previous constructions give the set of *ground terms* $T_{\Sigma,\Pi}(\emptyset)$, denoted by $T_{\Sigma,\Pi}$ and called *Herbrand universe* for $\langle\Sigma,\Pi\rangle$, and the set of *ground atoms* $B_{\Sigma,\Pi}(\emptyset)$, denoted by $B_{\Sigma,\Pi}$ and called *Herbrand base* for $\langle\Sigma,\Pi\rangle$.

Definition 1.4 *(Horn Clauses, Goals and Programs)*
Fixed a signature (Σ,Π) *and a set of variables* X*,* definite clauses*,* goals *or* queries *and* definite programs *on* (Σ,Π) *with variables* X *are, respectively, formulas of the type* $A \leftarrow B_1,\ldots,B_n$*, formulas of the type* $\leftarrow B_1,\ldots,B_n$*, where* A*,* $B_1,\ldots,B_n$ *are atoms in* $B_{\Sigma,\Pi}(X)$*, and sets of definite clauses.* ✓

1.2 Interpretations

In the previous section, we have defined in a purely syntactic way what a logic program is. The first thing we need to start talking about semantics is an interpretation for the symbols which constitute the program. We will identify interpretations with the category of the algebras ([6, 7], for an excellent survey see [9]) whose signature is program's one. Let us begin by recalling the basic definitions of Category Theory ([15, 5]).

Definition 1.5 *(Graphs)*
A graph *is a structure* $(dom, cod\colon A \to O)$*, where* A *and* O *are classes*[1] *of, respectively, arrows and objects, and* dom *and* cod *are functions which associate to each arrow, respectively, a domain and a codomain.* ✓

Given a graph G, the class of its *composable arrows* is

$$A \times_O A = \Big\{\langle g, f\rangle \;\Big|\; g, f \in A \text{ and } dom(g) = cod(f)\Big\}.$$

A category is a graph where each object has an identity arrow and arrows are closed under a given operation of composition.

[1] We shall not worry about foundational problems. We suppose to be working in a given Grothendieck universe.

$$
\begin{array}{ccc}
|\mathcal{A}|^n & \xrightarrow{\sigma_\mathcal{A}} & |\mathcal{A}| \\
\phi^n \downarrow & & \downarrow \phi \\
|\mathcal{B}|^n & \xrightarrow{\sigma_\mathcal{B}} & |\mathcal{B}|
\end{array}
\qquad\qquad
\begin{array}{ccc}
\rho_\mathcal{A} & & \\
\phi \downarrow & & \\
\phi(\rho_\mathcal{A}) & \hookrightarrow & \rho_\mathcal{B}
\end{array}
$$

Figure 1: (Σ, Π)-homomorphisms

Definition 1.6 *(Categories)*
A category $\mathbf{C}$ *is a graph together with two additional functions*

$$id \colon O \to A \text{ and } \circ \colon A \times_O A \to A,$$

called, respectively, identity and composition, such that

$$\forall \mathcal{A} \in O,\ cod\Big(id(\mathcal{A})\Big) = \mathcal{A} = dom\Big(id(\mathcal{A})\Big),$$

$$\forall \langle g, f \rangle \in A \times_O A,\ cod\Big(g \circ f\Big) = cod(g) \text{ and } dom\Big(g \circ f\Big) = dom(f).$$

Moreover, $\circ$ *is associative and for all* $f \in A$*, given* $\mathcal{A} = dom(f)$ *and* $\mathcal{B} = cod(f)$*, we have* $f \circ id(\mathcal{A}) = f = id(\mathcal{B}) \circ f$*.* ✓

Usually, the arrows of a category, also called *morphisms*, are denoted by $f \colon \mathcal{A} \to \mathcal{B}$, where $\mathcal{A} = dom(f)$ and $\mathcal{B} = cod(f)$ and identities by $id_\mathcal{A}$. The class of arrows $f \colon \mathcal{A} \to \mathcal{B}$ in $\mathbf{C}$ is indicated as $\mathbf{C}[\mathcal{A}, \mathcal{B}]$. Moreover, in dealing with a category $\mathbf{C}$ the actual classes A and O are never mentioned: we write $\mathcal{A} \in \mathbf{C}$ for objects and f in $\mathbf{C}$ for arrows.

A *subcategory* $\mathbf{B}$ of $\mathbf{C}$ is a category whose classes of objects and arrows are contained in the respective classes of $\mathbf{C}$; $\mathbf{B}$ is *full* when for each $\mathcal{A}, \mathcal{B} \in \mathbf{B}$ we have $\mathbf{B}[\mathcal{A}, \mathcal{B}] = \mathbf{C}[\mathcal{A}, \mathcal{B}]$.

Definition 1.7 *(*$\mathbf{Alg}_{\Sigma,\Pi}$*)*
A (Σ, Π)-algebra $\mathcal{A}$ *consists of*

(i) *a set* A*, called* carrier *of the algebra and denoted by* $|\mathcal{A}|$*;*

(ii) *for each* $n \in \mathbb{N}$ *and* $\sigma \in \Sigma_n$ *an operation* $\sigma_\mathcal{A} \colon A^n \to A$*;*

(iii) *for each* $n \in \mathbb{N}$ *and* $\rho \in \Pi_n$ *a predicate* $\rho_\mathcal{A} \subseteq A^n$*.*

A (Σ, Π)-homomorphism *between the* (Σ, Π)*-algebras* $\mathcal{A}$ *and* $\mathcal{B}$ *is a function* $\phi \colon |\mathcal{A}| \to |\mathcal{B}|$ *which respects operations and predicates (see Figure 1), i.e., such that:*

(i) *for each* $n \in \mathbb{N}$ *and* $\sigma \in \Sigma_n$, $\phi\Big(\sigma_\mathcal{A}(a_1, \ldots, a_n)\Big) = \sigma_\mathcal{B}\Big(\phi(a_1), \ldots, \phi(a_n)\Big)$*;*

(ii) *for each* $n \in \mathbb{N}$ *and* $\rho \in \Pi_n$, $\phi^n(\rho_\mathcal{A}) \subseteq \rho_\mathcal{B}$*,*

where ϕ^n *is the cartesian product of* n *copies of* ϕ*.*

This, with the usual notion of composition for homomorphisms, gives the category $\mathbf{Alg}_{\Sigma,\Pi}$*.* ✓

In order to simplify notation, we will denote ϕ^n by ϕ itself. Moreover, we will use $\pi\mathcal{A}$ to indicate the set $\{\rho(e_1, \ldots, e_n) \,|\, (e_1, \ldots, e_n) \in \rho_\mathcal{A},\ \rho \in \Pi\}$ and we extend the notation $[\![\_]\!]_\rho$ to subsets of $\pi\mathcal{A}$ for $\mathcal{A} \in \mathbf{Alg}_{\Sigma,\Pi}$.

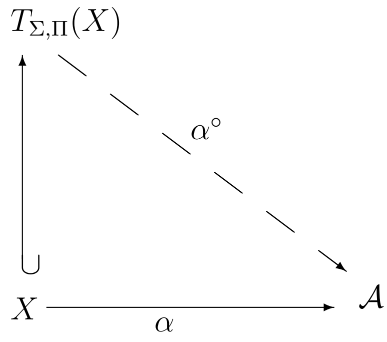

Figure 2: Free algebra on a set of generators X

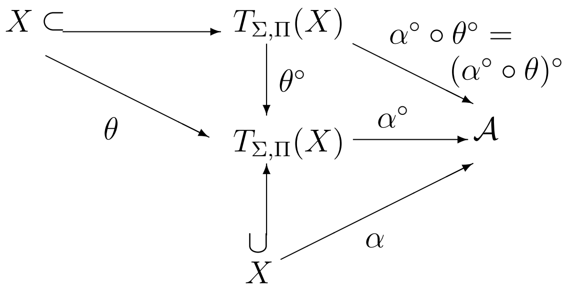

Figure 3: Composition of Substitutions

The set $T_{\Sigma,\Pi}(X)$ given in the previous section can be given the structure of a (Σ, Π)-algebra by defining operations and predicates as follows:

$$\text{for } \sigma \in \Sigma_n,\ \sigma_{T_{\Sigma,\Pi}(X)}(t_1, \ldots, t_n) = \sigma(t_1, \ldots, t_n) \quad \text{and} \quad \text{for } \rho \in \Pi,\ \rho_{T_{\Sigma,\Pi}(X)} = \emptyset.$$

The same construction makes $T_{\Sigma,\Pi}$ be a (Σ, Π)-algebra. Observe that variables in $T_{\Sigma,\Pi}(X)$ are not considered operations of arity zero, but just elements of the algebra. This allows (Σ, Π)-homomorphisms to map variables to any element of the target algebras and, therefore, to capture the notions of *substitutions* and *evaluations*.

Proposition 1.8 *($T_{\Sigma,\Pi}(X)$ is the Free Algebra on X)*
Let $\mathcal{A}$ be a (Σ, Π)-algebra. Given an assignment *of values in $\mathcal{A}$ to the variables in X, i.e., a function $\alpha: X \to \mathcal{A}$, there exists a unique (Σ, Π)-homomorphism $\alpha^\circ: T_{\Sigma,\Pi}(X) \to \mathcal{A}$ which extends α (see Figure 2), i.e., such that $(\alpha^\circ)_{|X} = \alpha$.* ✓

The (Σ, Π)-homomorphisms obtained as liftings of assignments are called *evaluation*. An assignment $\theta: X \to T_{\Sigma,\Pi}(X)$ and the correspondent θ° are what is usually called substitution.

Evaluations can be composed by composing their liftings as homomorphism. For instance, the composition of $\theta: X \to T_{\Sigma,\Pi}(X)$ and $\alpha: X \to \mathcal{A}$ gives rise to the assignment $(\alpha^\circ \circ \theta^\circ)_{|X} = \alpha^\circ \circ \theta: X \to \mathcal{A}$, as can be seen in Figure 3. In the following, with abuse of notation, we will forget any difference between α and α°. We will write αt to denote the evaluation of t under the assignment α, i.e., $\alpha^\circ(t)$. Moreover, composition of evaluations will be denoted by left juxtaposition. Therefore, for instance, we will write $\alpha\theta(t_1, \ldots, t_n)$ for

$$\Big(\alpha^\circ \circ \theta^\circ(t_1), \ldots, \alpha^\circ \circ \theta^\circ(t_n)\Big) = \Big((\alpha^\circ \circ \theta)^\circ(t_1), \ldots, (\alpha^\circ \circ \theta)^\circ(t_n)\Big).$$

We need a formal definition to establish whether a formula holds under an interpretation $\mathcal{A}$. This is the purpose of the next definition, in which $\mathcal{A}$ denotes a (Σ, Π)-algebra, α a generic assignment $T_{\Sigma,\Pi}(X) \to \mathcal{A}$, $A, B_1, \ldots, B_n$ are atoms, C ranges over clauses and P over programs.

Definition 1.9 *(Satisfaction Relation)*
The satisfaction relation $\models$ *between* (Σ, Π)*-algebras, and clauses, programs and goals is the smallest relation such that:*

$$\begin{array}{lll}
\mathcal{A} \models_\alpha \rho(t_1, \ldots, t_n) & \text{if and only if} & (\alpha t_1, \ldots, \alpha t_n) \in \rho_{\mathcal{A}}; \\
\mathcal{A} \models_\alpha B_1, \ldots, B_n & \text{if and only if} & \mathcal{A} \models_\alpha B_i \text{ for } i = 1, \ldots, n; \\
\mathcal{A} \models_\alpha A \leftarrow B_1, \ldots, B_n & \text{if and only if} & \mathcal{A} \models_\alpha B_1, \ldots, B_n \Rightarrow \mathcal{A} \models_\alpha A; \\
\mathcal{A} \models A \leftarrow B_1, \ldots, B_n & \text{if and only if} & \mathcal{A} \models_\alpha A \leftarrow B_1, \ldots, B_n \text{ for each } \alpha; \\
\mathcal{A} \models P & \text{if and only if} & \mathcal{A} \models C \text{ for each } C \in P; \\
\mathcal{A} \models \leftarrow B_1, \ldots, B_n & \text{if and only if} & \mathcal{A} \models_\alpha B_1, \ldots, B_n \text{ for some } \alpha.
\end{array}$$
✓

Models are those interpretations under which the logical implications specified by the clauses in the program are all realized. Some of the facts holding in a model of P will be logical consequences of the program, while other will just depend on the nature of the particular model. Hence, the consequences of a program, i.e., its semantics, are defined to be the set of facts which hold under every model of P. That is formally stated in the next definitions.

Definition 1.10 *(Models of Programs)*
The category of the models *of a program P, denoted by* $\mathbf{Mod}(P)$*, is the full subcategory of* $\mathbf{Alg}_{\Sigma,\Pi}$ *consisting of those algebras which satisfy P, i.e., the algebras $\mathcal{M}$ such that $\mathcal{M} \models P$.* ✓

Definition 1.11 *(Logical Consequences)*
An atom A is a logical consequence *of a program P, written $P \models A$, if and only if $\mathcal{M} \models A$ for any $\mathcal{M} \in \mathbf{Mod}(P)$.* ✓

1.3 Fixed-Point Semantics

Classical Logic Programming theory is concerned with the study of different characterizations of the set of logical consequences of programs. One of the classical approaches is the so-called fixed-point semantics, in which such a set is constructed as least fixed-point of a endofunction on the set of subsets of ground atoms. In this section we recall this approach.

Definition 1.12 *(Consequence Operators)*
Given a program P, the family of the immediate consequence operators *is*

$$\mathcal{T}_P = \left\{ \mathcal{T}_{\mathcal{A}} \colon 2^{\pi\mathcal{A}} \to 2^{\pi\mathcal{A}} \mid \mathcal{A} \in \mathbf{Alg}_{\Sigma,\Pi} \right\},$$

where $\mathcal{T}_{\mathcal{A}}$ is the function which maps $B \subseteq \pi\mathcal{A}$ to the set

$$\left\{ \rho(\theta t_1, \ldots, \theta t_n) \;\middle|\; \begin{array}{l} \theta \colon X \to \mathcal{A},\; \rho^i(\theta t^i_1, \ldots, \theta t^i_{n_i}) \in B,\; i = 1, \ldots, k, \\ \rho(t_1, \ldots, t_n) \leftarrow \rho(t^1_1, \ldots, t^1_{n_1}), \ldots, \rho(t^k_1, \ldots, t^k_{n_k}) \in P \end{array} \right\} \cup B.$$

Since $\bigcup_n \mathcal{T}^n_{\mathcal{A}}(B)$ exists for each $B \subseteq \pi\mathcal{A}$, where $\mathcal{T}^n_{\mathcal{A}}(B)$ denotes n nested applications of $\mathcal{T}_{\mathcal{A}}$ to B and $\bigcup$ is the union of sets, we can define the function $\mathcal{T}^\omega_{\mathcal{A}}$ which maps B to $\bigcup_n \mathcal{T}^n_{\mathcal{A}}(B)$ and the family of consequence operators $\mathcal{T}^\omega_P = \left\{ \mathcal{T}^\omega_{\mathcal{A}} \;\middle|\; \mathcal{A} \in \mathbf{Alg}_{\Sigma,\Pi} \right\}$. ✓

Models built from term algebras are called Herbrand models. The following is a well-known fact about Herbrand models which does not hold for general models.

Proposition 1.13 *(Characterization of Herbrand Models)*
A term algebra $\mathcal{A}$ is a model of P if and only if $T_{\mathcal{A}}(\pi\mathcal{A}) \subseteq \pi\mathcal{A}$. ✓

Definition 1.14 *(Least Herbrand Model)*
The least Herbrand model *of a program P is the algebra $T_{\Sigma,\Pi,P}$ obtained from $T_{\Sigma,\Pi}$ by enforcing $(t_1,\ldots,t_n) \in \rho_{T_{\Sigma,\Pi,P}}$ if and only if $\rho(t_1,\ldots,t_n) \in T^{\omega}_{T_{\Sigma,\Pi}}(\emptyset)$.* ✓

The least Herbrand model is the most relevant model in that it completely characterizes the semantics of a program: the facts true in $T_{\Sigma,\Pi,P}$ are exactly the consequences of P, i.e., the facts true in every model. In this sense the least Herbrand model is universal among the models P. In the next section, we will see that it is universal also in some precise algebraic senses. The standard results concerning $T_{\Sigma,\Pi,P}$ are listed in the following propositions.

Proposition 1.15 *($T_{\Sigma,\Pi,P}$ is Universal)*
$T_{\Sigma,\Pi,P}$ is a model of P and $T_{\Sigma,\Pi,P} \models A \Leftrightarrow P \models A$. ✓

Proposition 1.16 *(Herbrand's Theorem)*
$T_{\Sigma,\Pi,P} \models \leftarrow B_1,\ldots,B_n$ if and only if $\mathcal{M} \models \leftarrow B_1,\ldots,B_n$ for each $\mathcal{M} \in \mathbf{Mod}(P)$. ✓

2 Categorical Semantics

In this section, we give two categorical characterizations of the least Herbrand model. In particular, we show that the classical notion of algebraic structure over a category can be applied to case of Logic Programming getting the category of Herbrand models from the category of Herbrand interpretations via the standard construction of *algebras for endofunctors* (see [15, 4, 5]). As a consequence, we find $T_{\Sigma,\Pi,P}$ as the least fixed-point of an endofunctor.

Moreover, we show that the category of (general) models is *reflective* in the category of interpretations and we find again the least Herbrand model as reflection of the Herbrand interpretation under which no fact holds.

We start by recalling that $T_{\Sigma,\Pi,P}$ is the initial object in $\mathbf{Mod}(P)$. For a discussion about the relevance of concept of initiality see, for instance, [9].

Definition 2.1 *(Initiality)*
An object $\Im$ in a category $\mathbf{C}$ is initial *in $\mathbf{C}$ if $\Im$ belongs to $\mathbf{C}$ and for any $c \in \mathbf{C}$ there exists a unique morphism from $\Im$ to c.* ✓

It is worthwhile observing that the initial object in $\mathbf{Alg}_{\Sigma,\Pi}$ is $T_{\Sigma,\Pi}$, the free algebra over the empty set of generators.

The universality of the initial object is reflected in the following standard result from Category Theory.

Proposition 2.2 *(Uniqueness of the Initial Object)*
The initial object of a category $\mathbf{C}$*, if it exists, is unique up to isomorphisms.* ✓

Proposition 2.3 *(Initiality of* $T_{\Sigma,\Pi,P}$*)*
$T_{\Sigma,\Pi,P}$ *is initial in* $\mathbf{Mod}(P)$*.*

Proof. Since $T_{\Sigma,\Pi,P}$, when we forget about predicates, coincides with the initial (Σ,Π)-algebra $T_{\Sigma,\Pi}$, given any $\mathcal{M} \in \mathbf{Mod}$, it exists exactly one function ϕ from $T_{\Sigma,\Pi,P}$ to $\mathcal{M}$ which respects condition *(i)* of Definition 1.7. From Proposition 1.15, it is immediate to see that ϕ respects also condition *(ii)* and that, therefore, is the unique (Σ,Π)-homomorphism from $T_{\Sigma,\Pi,P}$ to $\mathcal{M}$. ✓

2.1 $T_{\Sigma,\Pi,P}$ as Endofunctor Fixed-Point

Definition 2.4 *(Endofunctors)*
Given two categories $\mathbf{C}$ *and* $\mathbf{D}$*, a* functor $\mathcal{F}\colon \mathbf{C} \to \mathbf{D}$ *is a function which maps objects in* $\mathbf{C}$ *to objects in* $\mathbf{D}$ *and morphisms in* $\mathbf{C}$ *to morphisms in* $\mathbf{D}$ *in such a way that:*

(i) $\mathcal{F}(h)\colon \mathcal{F}(\mathcal{A}) \to \mathcal{F}(\mathcal{B})$*, for each* $h\colon \mathcal{A} \to \mathcal{B}$ *in* $\mathbf{C}$*;*

(ii) $\mathcal{F}(id_{\mathcal{A}}) = id_{\mathcal{F}(\mathcal{A})}$*;*

(iii) $\mathcal{F}(k \circ h) = \mathcal{F}(k) \circ \mathcal{F}(h)$*.*

An endofunctor *is a functor from a category to the category itself.* ✓

Given a category $\mathbf{C}$, the function which is the identity both on objects and arrows is clearly an endofunctor. It will be denoted by $\mathbf{1_C}$. Composition of functors is defined in the obvious way and denoted by left juxtaposition.

Definition 2.5 *(*$\mathcal{H}\mathbf{Alg}_{\Sigma,\Pi}$*)*
Let $\mathcal{H}\mathbf{Alg}_{\Sigma,\Pi}$ *be the full subcategory of* $\mathbf{Alg}_{\Sigma,\Pi}$ *consisting of those algebras which are Herbrand interpretations, i.e., the* (Σ,Π)*-algebras of terms.* ✓

In the following, we will denote by $\mathcal{H}\mathbf{Mod}(P)$ the full subcategory of $\mathcal{H}\mathbf{Alg}_{\Sigma,\Pi}$ consisting of the (Herbrand) models of P.

Now, we can see that the immediate consequence operator $\boldsymbol{\mathcal{T}}_{T_{\Sigma,\Pi}}$ gives an endofunctor on the category of Herbrand interpretations.

Proposition 2.6 *(Functor* $\mathcal{T}_P$*)*
Given a program P *on* $\langle\Sigma,\Pi\rangle$*, let us consider the function* $\mathcal{T}_P\colon \mathcal{H}\mathbf{Alg}_{\Sigma,\Pi} \to \mathcal{H}\mathbf{Alg}_{\Sigma,\Pi}$ *defined by:*

(i) $\mathcal{T}_P(\mathcal{A})$ *has the same elements of* $\mathcal{A}$*;*

(ii) $\sigma_{\mathcal{T}_P(\mathcal{A})} = \sigma_{\mathcal{A}}$ *for each* $\sigma \in \Sigma$*;*

(iii) $\rho_{\mathcal{T}_P(\mathcal{A})} = [\![\boldsymbol{\mathcal{T}}_{T_{\Sigma,\Pi}}(\pi\mathcal{A})]\!]_\rho$ *for each* $\rho \in \Pi$*;*

(iv) $\mathcal{T}_P$ *is the identity on homomorphisms.*

Then $\mathcal{T}_P$ *is an endofunctor.*

Proof. Properties *(ii)* and *(iii)* in Definition 2.4 obviously hold. In order to show *(i)*, since h is necessarily the identity,it is enough to show that

$$\Big(\forall \rho \in \Pi,\ \rho_{\mathcal{A}} \subseteq \rho_{\mathcal{B}}\Big) \Rightarrow \Big(\forall \rho \in \Pi,\ \rho_{\mathcal{T}_P(\mathcal{A})} \subseteq \rho_{\mathcal{T}_P(\mathcal{B})}\Big).$$

Let $(t_1, \ldots, t_n) \in \rho_{\mathcal{T}_P(\mathcal{A})}$. Then, there exist $(t_1^i, \ldots, t_{n_i}^i) \in \rho_{\mathcal{A}}^i,\ i = 1, \ldots, k$ and

$$\rho(T_1, \ldots, T_n) \leftarrow \rho^1(T_1^1, \ldots, T_{n_1}^1), \ldots, \rho^k(T_1^k, \ldots, T_{n_k}^k) \in P$$

with $\theta: X \to \mathcal{A}$ such that $\theta T_i = t_i$ and $\theta T_i^k = t_i^k$, $i = 1, \ldots, k$.
Since by hypothesis $h(\rho_{\mathcal{A}}^i) \subseteq \rho_{\mathcal{B}}^i$, we have $h\theta(T_1^i, \ldots, T_{n_i}^i) \in \rho_{\mathcal{B}}^i$, $i = 1, \ldots, k$ and, therefore, $h\theta(T_1, \ldots, T_n) = (t_1, \ldots, t_n) \in \rho_{\mathcal{T}_P(\mathcal{B})}$. ✓

Now, let us recall the basic definitions about algebras for endofunctors as stated in [5].

Definition 2.7 *($(\mathcal{T}{:}\,\mathbf{C})$)*
Given a category $\mathbf{C}$ *and an endofunctor* $\mathcal{T}{:}\,\mathbf{C} \to \mathbf{C}$*, a* $\mathcal{T}$-algebra *on* $\mathbf{C}$ *is a pair* $(\mathcal{A}, a)$*, where* $\mathcal{A}$ *is an object in* $\mathbf{C}$ *and* $a{:}\,\mathcal{T}(\mathcal{A}) \to \mathcal{A}$ *is a morphism in* $\mathbf{C}$*.*

A $\mathcal{T}$-homomorphism $\phi{:}\,(\mathcal{A}, a) \to (\mathcal{B}, b)$ *is a morphism* $\phi{:}\,\mathcal{A} \to \mathcal{B}$ *in* $\mathbf{C}$ *such that* $\phi \circ a = b \circ \mathcal{T}(\phi)$*, i.e., such that the following diagram commutes.*

$$\begin{array}{ccc}
\mathcal{T}(\mathcal{A}) & \xrightarrow{a} & \mathcal{A} \\
{\scriptstyle \mathcal{T}(\phi)}\big\downarrow & & \big\downarrow{\scriptstyle \phi} \\
\mathcal{T}(\mathcal{B}) & \xrightarrow{b} & \mathcal{B}
\end{array}$$

This defines $(\mathcal{T}{:}\,\mathbf{C})$*, the category of* $\mathcal{T}$*-algebras on* $\mathbf{C}$*.* ✓

Definition 2.8 *(Fixed-Points)*
A fixed-point *for* $\mathcal{T}$ *is a* $\mathcal{T}$*-algebra* $(\mathcal{A}, a) \in (\mathcal{T}{:}\,\mathbf{C})$ *such that* $a{:}\,\mathcal{T}(\mathcal{A}) \to \mathcal{A}$ *is an isomorphism.* ✓

Hence, a fixed-point for an endofunctor is a $\mathcal{T}$-algebra $\mathcal{A}$ such that $\mathcal{A} \cong \mathcal{T}(\mathcal{A})$. It is therefore clear as this represents a generalization of the concept of fixed-point in the categorical language, where everything is treated up to isomorphisms.

Proposition 2.9 *(lfp(T))*
If $(\mathcal{T}{:}\,\mathbf{C})$ *admits an initial object* $(\mathcal{A}, a)$*, then* $(\mathcal{A}, a)$ *is a fixed-point for* $\mathcal{T}$*.*

Proof. If $(\mathcal{A}, a)$ is initial, then $\exists! \phi{:}\,(\mathcal{A}, a) \to (\mathcal{T}(\mathcal{A}), \mathcal{T}(a))$ and $id_{\mathcal{A}}$ is the unique morphism from $(\mathcal{A}, a)$ to itself. Then the following diagram commutes.

$$\begin{array}{ccc}
\mathcal{T}(\mathcal{A}) & \xrightarrow{a} & \mathcal{A} \\
{\scriptstyle \mathcal{T}(\phi)}\big\downarrow & & \big\downarrow{\scriptstyle \phi} \\
\mathcal{T}^2(\mathcal{A}) & \xrightarrow{\mathcal{T}(a)} & \mathcal{T}(\mathcal{A}) \\
{\scriptstyle \mathcal{T}(a)}\big\downarrow & & \big\downarrow{\scriptstyle a} \\
\mathcal{T}(\mathcal{A}) & \xrightarrow{a} & \mathcal{A}
\end{array}$$

The upper square commutes by definition of morphism in $(\mathcal{T}\colon \mathbf{C})$. Reading the whole rectangle, we have that $a \circ \phi$ is a $\mathcal{T}$-homomorphism from $(\mathcal{A}, a)$ to itself. Therefore, $a \circ \phi = id_{\mathcal{A}}$ and so $\mathcal{T}(a \circ \phi) = \mathcal{T}(a) \circ \mathcal{T}(\phi) = id_{\mathcal{T}(\mathcal{A})}$. On the other hand, from the upper square we have that $\mathcal{T}(a) \circ \mathcal{T}(\phi) = \phi \circ a$, from which $\phi \circ a = id_{\mathcal{T}(\mathcal{A})}$. Hence ϕ is an isomorphism. ✓

The following definition is now completely natural. The reader can find in [5] some examples which further justify it.

Definition 2.10 *(Least Fixed-Point)*
If $(\mathcal{T}\colon \mathbf{C})$ has an initial object, then it is called least fixed-point *of $\mathcal{T}$.* ✓

Let us consider now $(\mathcal{T}_P\colon \mathcal{H}\mathbf{Alg}_{\Sigma,\Pi})$. We will show that it has an initial object and that it coincides with $T_{\Sigma,\Pi,P}$. In other words, the (categorical) fixed-point of $\mathcal{T}_P$ is the least Herbrand model of P.

Proposition 2.11 *(Herbrand models vs $\mathcal{T}_P$-algebras, part 1)*
$\mathcal{A}$ is an Herbrand model of P if and only if $(\mathcal{A}, id_{\mathcal{A}}) \in (\mathcal{T}_P\colon \mathcal{H}\mathbf{Alg}_{\Sigma,\Pi})$.

Proof. Trivially from Proposition 1.13, since both conditions are equivalent to $\rho_{\mathcal{T}_P(\mathcal{A})} \subseteq \rho_{\mathcal{A}}$ for each $\rho \in \Pi$. ✓

Proposition 2.12 *(Herbrand models vs $\mathcal{T}_P$-algebras, part 2)*
Let $(\mathcal{A}, a)$, $(\mathcal{B}, b)$ be in $(\mathcal{T}_P\colon \mathcal{H}\mathbf{Alg}_{\Sigma,\Pi})$. Then ϕ is a (Σ, Π)-homomorphism from $\mathcal{A}$ to $\mathcal{B}$ if and only if it is a $\mathcal{T}_P$-homomorphism from $(\mathcal{A}, a)$ to $(\mathcal{B}, b)$.

Proof. $(\Leftarrow)$. Obvious.
$(\Rightarrow)$. It is evident that the diagram in Definition 2.7 commutes, because $\mathcal{T}_P(\mathcal{A})$ and $\mathcal{A}$ are term algebras obtained from $T_{\Sigma,\Pi,P}$, and therefore the unique (Σ, Π)-homomorphism between them, if any, is the identity on terms. ✓

Two categories $\mathbf{C}$ and $\mathbf{D}$ are said isomorphic if there exists a pair of functors $\mathcal{F}\colon \mathbf{C} \to \mathbf{D}$ and $\mathcal{G}\colon \mathbf{D} \to \mathbf{C}$ such that $\mathcal{G}\mathcal{F} = \mathbf{1_C}$ and $\mathcal{F}\mathcal{G} = \mathbf{1_D}$. Clearly, isomorphic categories are essentially the same.

Corollary 2.13 *($\mathcal{T}_P$-algebras are models)*
$(\mathcal{T}_P\colon \mathcal{H}\mathbf{Alg}_{\Sigma,\Pi}) \cong \mathcal{H}\mathbf{Mod}(P)$.

Proof. It follows immediately from Proposition 2.12 that the functor $\mathcal{F}\colon \mathcal{H}\mathbf{Mod}(P) \to (\mathcal{T}_P\colon \mathcal{H}\mathbf{Alg}_{\Sigma,\Pi})$ which sends $\mathcal{M}$ in $(\mathcal{M}, id_{\mathcal{M}})$ and is the identity on morphisms is an isomorphism of categories. ✓

Since isomorphisms of categories in particular are isomorphisms of classes of arrows between correspondent objects, it is immediate to see that they send initial objects to initial objects. The next corollary follows from this observation.

Corollary 2.14 *($T_{\Sigma,\Pi,P}$ is $lfp(\mathcal{T}_P)$)*
$T_{\Sigma,\Pi,P}$ is the least fixed-point of $\mathcal{T}_P$. ✓

2.2 Models are reflective in Interpretations

In this section, we present a characterization of the least Herbrand model based on an *adjunction* between interpretations and models.

Adjunctions, introduced in [12], provide an elegant way to formulate properties of free objects and universal constructions.

Definition 2.15 *(Adjunctions)*
An adjunction *from* $\mathbf{C}$ *to* $\mathbf{D}$ *is a triple* $\langle \mathcal{F}, \mathcal{G}, \varphi \rangle : \mathbf{C} \rightharpoonup \mathbf{D}$*, where* $\mathcal{F}: \mathbf{C} \to \mathbf{D}$ *and* $\mathcal{G}: \mathbf{D} \to \mathbf{C}$ *are functors and* φ *is a function which assigns to each pair of objects* $\mathcal{A} \in \mathbf{C}$ *and* $\mathcal{B} \in \mathbf{D}$ *a bijection*

$$\varphi_{\mathcal{A},\mathcal{B}}: \mathbf{D}[\mathcal{F}(\mathcal{A}), \mathcal{B}] \cong \mathbf{C}[\mathcal{A}, \mathcal{G}(\mathcal{B})],$$

which is natural both in $\mathcal{A}$ *and* $\mathcal{B}$*, i.e., such that for all* $k: \mathcal{A}' \to \mathcal{A}$ *and* $h: \mathcal{B} \to \mathcal{B}'$ *the following diagrams commute.*

$$\begin{array}{ccc} \mathbf{D}[\mathcal{F}(\mathcal{A}), \mathcal{B}] & \xrightarrow{\varphi_{\mathcal{A},\mathcal{B}}} & \mathbf{C}[\mathcal{A}, \mathcal{G}(\mathcal{B})] \\ {\scriptstyle \_ \circ \mathcal{F}(k)} \downarrow & & \downarrow {\scriptstyle \_ \circ k} \\ \mathbf{D}[\mathcal{F}(\mathcal{A}'), \mathcal{B}] & \xrightarrow{\varphi_{\mathcal{A}',\mathcal{B}}} & \mathbf{C}[\mathcal{A}', \mathcal{G}(\mathcal{B})] \end{array} \qquad \begin{array}{ccc} D[\mathcal{F}(\mathcal{A}), \mathcal{B}] & \xrightarrow{\varphi_{\mathcal{A},\mathcal{B}}} & \mathbf{C}[\mathcal{A}, \mathcal{G}(\mathcal{B})] \\ {\scriptstyle h \circ \_} \downarrow & & \downarrow {\scriptstyle \mathcal{G}(h) \circ \_} \\ \mathbf{D}[\mathcal{F}(\mathcal{A}), \mathcal{B}'] & \xrightarrow{\varphi_{\mathcal{A},\mathcal{B}'}} & \mathbf{C}[\mathcal{A}, \mathcal{G}(\mathcal{B}')] \end{array}$$

✓

If $\langle \mathcal{F}, \mathcal{G}, \varphi \rangle : \mathbf{C} \rightharpoonup \mathbf{D}$ is an adjunction, $\mathcal{F}$ is called *left adjoint* to $\mathcal{G}$ and, viceversa, $\mathcal{G}$ is called *right adjoint* to $\mathcal{F}$.

A well-know result in Category Theory is that left adjoints preserve initiality, i.e., they map initial objects to initial objects.

Definition 2.16 *(Reflections)*
A subcategory $\mathbf{C}$ *of* $\mathbf{D}$ *is said* reflective *in* $\mathbf{D}$ *if there exists a left adjoint, said* reflector *of* $\mathbf{D}$ *in* $\mathbf{C}$*, for the inclusion functor* $\mathbf{C} \hookrightarrow \mathbf{D}$*.* ✓

In the following, we will show the family of functions $\mathcal{T}^{\omega}_{P}$, defined in Section 1.3, defines a reflector of $\mathbf{Alg}_{\Sigma,\Pi}$ in $\mathbf{Mod}(P)$. Informally speaking, in our setting it means that any interpretation can be "completed" to be a model in a universal way, i.e., that any interpretation has a canonical representative—its reflection—in the category of models. The reader is referred to [15] for further considerations on the relevance the concept of reflection. By exploiting standard results from Category Theory, we show that the $T_{\Sigma,\Pi,P}$ is the canonical object linked by the adjunction to the empty Herbrand interpretation, i.e., $T_{\Sigma,\Pi}$.

Definition 2.17 *(Functor* $\mathcal{T}^{\omega}_{P}$*)*
Given a program P *on* $\langle \Sigma, \Pi \rangle$*, let us consider the function* $\mathcal{T}^{\omega}_{P}: \mathbf{Alg}_{\Sigma,\Pi} \to \mathbf{Alg}_{\Sigma,\Pi}$ *defined by:*

(i) $\mathcal{T}^{\omega}_{P}(\mathcal{A})$ *has the same elements of* $\mathcal{A}$*;*

(ii) $\sigma_{\mathcal{T}^{\omega}_{P}(\mathcal{A})} = \sigma_{\mathcal{A}}$ *for each* $\sigma \in \Sigma$*;*

(iii) $\rho_{\mathcal{T}^{\omega}_{P}(\mathcal{A})} = [\![\mathcal{T}^{\omega}_{\mathcal{A}}(\pi\mathcal{A})]\!]_{\rho}$ *for each* $\rho \in \Pi$*;*

(iv) $\mathcal{T}^{\omega}_{P}$ *is the identity on homomorphisms.* ✓

Proposition 2.18 *($\mathcal{T}_P^\omega$: $\mathbf{Alg}_{\Sigma,\Pi} \to \mathbf{Mod}(P)$)*
For each $\mathcal{A} \in \mathbf{Alg}_{\Sigma,\Pi}$, $\mathcal{T}_P^\omega(\mathcal{A})$ is a model. Moreover, $\phi: \mathcal{A} \to \mathcal{B}$ in $\mathbf{Alg}_{\Sigma,\Pi}$ if and only if $\phi: \mathcal{T}_P^\omega(\mathcal{A}) \to \mathcal{T}_P^\omega(\mathcal{B})$ is in $\mathbf{Mod}(P)$. ✓

Proof. Obviously, since $\mathcal{T}_{\mathcal{A}}^\omega(\pi\mathcal{A}) = \mathcal{T}_{\mathcal{A}}^\omega(\mathcal{T}_{\mathcal{A}}^\omega(\pi\mathcal{A}))$, we have that $\mathcal{T}_P^\omega(\mathcal{A}) = \mathcal{T}_P^{(\omega)}(\mathcal{T}_P^{(\omega)}(\mathcal{A}))$ which, therefore, is a model.

Concerning the claim about morphisms, the left implication is clearly true. In order to show the converse implication, suppose now that $(e_1, \ldots, e_n) \in \rho_{\mathcal{T}_P^\omega(\mathcal{A})}$. We show by induction on the least $k \in \mathbb{N}$ such that $\rho(e_1, \ldots, e_n) \in \mathcal{T}_{\mathcal{A}}^k(\pi\mathcal{A})$ that $\phi(e_1, \ldots, e_n) \in \rho_{\mathcal{T}_P^\omega(\mathcal{B})}$.

Induction base. In this case we have that $\rho(e_1, \ldots, e_n) \in \pi\mathcal{A}$.
Then we have that $\phi(e_1, \ldots, e_n) \in \rho_{\mathcal{T}_P^\omega(\mathcal{B})}$, since ϕ is a (Σ, Π)-homomorphism and $\pi\mathcal{B} \subseteq \pi\mathcal{T}_P^\omega(\mathcal{B})$.

Inductive step. There exists $k \in \mathbb{N} \setminus \{\nvdash\}$ such that for some $\theta: X \to \mathcal{A}$ and for some $\rho(t_1, \ldots, t_n) \leftarrow \rho^1(t_1^1, \ldots, t_{n_1}^1), \ldots, \rho^h(t_1^h, \ldots, t_{n_h}^h) \in P$, we have that $\theta t_i = e_i$ for $i = 1, \ldots, n$ and $\rho^1(t_1^1, \ldots, t_{n_1}^1), \ldots, \rho^h(t_1^h, \ldots, t_{n_h}^h) \in \mathcal{T}_{\mathcal{A}}^{(k-1)}(\pi\mathcal{A})$. Therefore, by a straightforward application of the inductive hypothesis, the proof is concluded. ✓

As a corollary to the previous proposition we have that $\mathcal{T}_P^{(\omega)}: \mathbf{Alg}_{\Sigma,\Pi} \to \mathbf{Mod}(P)$ is a functor.

Proposition 2.19 *($\langle \mathcal{T}_P^\omega, \hookleftarrow \rangle: \mathbf{Alg}_{\Sigma,\Pi} \rightharpoonup \mathbf{Mod}(P)$)*
$\mathcal{T}_P^{(\omega)}$ is the left adjoint to the inclusion functor of $\mathbf{Mod}(P)$ in $\mathbf{Alg}_{\Sigma,\Pi}$.

Proof. Given $\mathcal{A} \in \mathbf{Alg}_{\Sigma,\Pi}$ and $\mathcal{M} \in \mathbf{Mod}(P)$ we have that $\phi: \mathcal{A} \to \mathcal{M}$ if and only if $\phi: \mathcal{T}_P^{(\omega)}(\mathcal{A}) \to \mathcal{T}_P^{(\omega)}(\mathcal{M}) = \mathcal{M}$. Therefore the natural isomorphism we are seeking is the identity $\mathbf{Alg}_{\Sigma,\Pi}[\mathcal{A}, \mathcal{M}] = \mathbf{Mod}(P)[\mathcal{T}_P^\omega(\mathcal{A}), \mathcal{T}_P^\omega(\mathcal{M})] = \mathbf{Mod}(P)[\mathcal{T}_P^\omega(\mathcal{A}), \mathcal{M}]$. ✓

Finally, we find again the least Herbrand model as the image of the initial (Σ, Π)-algebra via $\mathcal{T}_P^\omega$.

Corollary 2.20 *($T_{\Sigma,\Pi,P}$ is the reflection of $T_{\Sigma,\Pi}$)*
$T_{\Sigma,\Pi,P} = \mathcal{T}_P^{(\omega)}(T_{\Sigma,\Pi})$.

Proof. $T_{\Sigma,\Pi}$ is initial in $\mathbf{Alg}_{\Sigma,\Pi}$ and $\mathcal{T}_P^\omega$ preserves initiality. ✓

Acknowledgments. *I wish to thank Roberto Di Meglio and Maurizio Gabbrielli for their valuable comments. Giorgio Levi deserves special thanks for his encouragement to write this note.*

References

[1] K.R. Apt. Introduction to Logic Programming. In *Handbook of Theoretical Computer Science*, vol. B, Elsevier and MIT Press, 1990.

[2] A. Asperti and A. Corradini. A categorical model for logic programs: indexed monoidal categories. In *Proceedings of REX '92*. To appear in the LNCS series, Springer-Verlag, 1993.

[3] A. Asperti and S. Martini. Projections instead of Variables, A Category Theoretic Interpretation of Logic Programs. In *Proceedings of the 6th International Conference on Logic Programming*, 1989.

[4] M. Barr and C. Wells. *Toposes, Triples and Theories*, Springer-Verlag, 1985.

[5] M. Barr and C. Wells. *Category Theory for Computing Science*, Prentice-Hall, 1990.

[6] G. Birkhoff. On the Structure of Abstract Algebras, in *Proceedings of the Cambridge Philosophical Society*, n. 31, 1935.

[7] G. Birkhoff and J. Lipson. Heterogeneous Algebras. *Journal of Combinatorial Theory*, n. 8, 1970.

[8] A. Corradini and U. Montanari. An Algebraic Semantics for Structured Transitions Systems and its application to Logic Programming. *Theoretical Computer Science*, n. 103, 1992.

[9] J. Goguen and J. Meseguer. Initiality, Induction, and Computability, in *Algebraic Methods in Semantics*, M. Nivat and J. Reynolds, Eds., Cambridge University Press, 1985.

[10] J. Goguen and J. Meseguer. Eqlog: Equality, Types, and Generic Modules for Logic Programming, in *Functional and Logic Programming*, Douglas DeGroot and Gary Lindstrom, Eds., Prentice-Hall, 1986.

[11] J. Goguen and J. Meseguer. Models and Equality for Logical Programming, in *Proceedings of TAPSOFT '87*, LNCS 250, Springer-Verlag, 1987.

[12] D.M. Kan. Adjoint Functors. *Transactions of the American Mathematical Society*, n. 87, 1958.

[13] R.A. Kowalski. Predicate Logic as a Programming Language, in *Proceedings of IFIP '74*, North-Holland, 1974.

[14] J.W. Lloyd. *Foundations of Logic Programming*, Second Edition, Springer-Verlag, 1987.

[15] S. MacLane. *Categories for the Working Mathematician*, Graduate Text in Mathematics, Springer-Verlag, 1971.

[16] M.J. Maher. *Semantics of Logic Programs*, Technical Report 85/14, Department of Computer Science, University of Melbourne, Australia, 1985.

[17] J. Meseguer. General Logics, in *Logical Colloquium '87*, North-Holland, 1987.