A SOUND AND EQUATIONALLY-COMPLETE DEDUCTION SYSTEM FOR PARTIAL CONDITIONAL (HIGHER-ORDER) TYPES

(Extended Abstract)

Maura Cerioli
Dipartimento di Matematica - Università di Genova
Via L.B. Alberti 4 - 16132 Genova Italy

Abstract. *Higher-order algebraic specifications of partial functions lead naturally to consider partial conditional specifications, ie partial specifications with axioms of the form* $\wedge_{i \in I} t_i = t'_i \supset t = t'$ *where on the left-hand side the equality is strong (the equality holds iff either both sides are defined and equal (existential equality) or both are undefined). Contrary to the well explored case of positive conditional axioms (only existential equalities on the left) those specifications do not always admit free models (the related theory is much more subtle and still unsettled).*
Relying on and completing some our previous results, we present in this paper a deduction system which is complete w.r.t. strong equalities between open terms, with an application to the existence of free objects. Since positive conditional partial specifications and conditional total specifications are special cases of the paradigm investigated here, the presented theory generalizes the related Birkhoff-like deduction theory.
The system we exhibit here looks nice since it handles also the case of infinitary conjunctions on the left of the axioms; it reduces to a classical one for the positive conditional case just dropping one rule, and finally solves the empty-carrier problem, noticed by Huet and Goguen-Meseguer, without using explicit quantification.

Introduction

Higher-order algebraic specifications are a natural and extremely useful extension of classical first-order algebraic specifications of abstract data types (see e.g. [11]). Since clearly we need to specify also partial functions, the partial algebra specifications framework is an obvious candidate for investigating the related problems. (Indeed it can be shown that the problems considered here are essentially the same in the total framework and our results apply to that case too).

Work partially supported by MPI 40%

The delicate point with higher-order algebraic specification is extensionality. Indeed classes of algebras satisfying axioms of the form

$$(\forall\ x \in Dom(f):\ f(x) = g(x)) \supset f = g$$

are not closed under subobjects, since the quantifications only involves the left-hand side. Thus it is necessary to require a more restrictive extensionality condition, which we call term-extensionality:

$$\wedge_{t \in T\Sigma}\ f(t) = g(t) \supset f = g.$$

This formula is a particular case of *conditional axiom*. Conditional axioms have the form $\wedge \Delta \supset \varepsilon$, where Δ is a possibly infinite set of both *strong* and existential equalities and ε is an equality too, and their models classes are closed under subobjects and isomorphisms.

Like the higher-order ones, in general partial conditional specifications do not admit initial models, contrary to the classical case (considered, e.g., in the work of Broy and Wirsing [6]; see also Burmeister [5], Reichel [12]) of positive conditional specifications, ie those whose axioms have only existential equalities in the premises.

In [1] we investigated the problem and gave necessary and sufficient conditions for the existence of initial models, related them to logical deduction and exhibited a sound system which is complete w.r.t. closed elementary formulas. In this paper we tackle the problem of giving a sound system complete w.r.t. *open* elementary formulas; ie we want to get the analogous of the classical Birkhoff's result for equational theories.

The first problem to solve is soundness. Goguen and Meseguer in [10] have shown that the usual one-sorted inference system trivially adapted to the many-sorted case produces unsound deductions whenever empty carriers are allowed, and proposed adding explicit quantification to the formulas, as classical logic does, to avoid the problem. In our formalism we can eliminate the unsound deductions without introducing explicit quantification; indeed the definedness predicate does the job: if x is a variable of sort s, then D(x) in the premises of an axiom guarantees the existence of an element of sort s. Thus we can treat the quantification implicitly, by representing a formula $\forall X: \wedge \Delta \supset \varepsilon$ by $\wedge (\{D(x) \mid x \in X\} \cup \Delta) \supset \varepsilon$.

The second problem is completeness. The completeness w.r.t. open equalities of an inference system is usually shown (see e.g. [10,14]) by proving that the relation $\equiv$ between open terms associated with the system (defined by $t \equiv t'$ iff "$t = t'$) is a congruence s.t. $Fr = T_\Sigma(X)/\equiv$ is a model of T, since Fr satisfies all and only the deduced existential equalities. But this proof is possible just because both positive and total conditional specifications always have free objects; indeed if $T_\Sigma(X)/\equiv$ is a model, then it is also the free object for X in the class of T models, because of the soundness of L(T). Therefore in the case of (non-positive) conditional specifications we cannot use this technique.

In [1] we have shown the completeness, w.r.t. the closed elementary formulas, of an inference system, in the following denoted by $CL_g(T)$, where g stands for ground. Here we generalize $CL_g(T)$ to a system $CL_v(T)$, complete w.r.t. open elementary formulas.

In section 1 we introduce the overall setting and a sound logical system; this system is complete for all positive conditional specifications w.r.t. open existential equalities, but is not complete w.r.t. partial conditional specifications. In section 2 we add to this system one rule, making it complete w.r.t. strong equality for all partial, possibly non-positive, conditional specifications. Finally in section 3 we relate this complete system and the existence of free objects. All the proofs are omitted and may be found in [2].

The theory of partial algebras is nowadays well established and widely used (see e.g. [7, 14, 3, 4]). Hence we briefly collect in appendix some basic notions about partial specifications just in order to fix the notation; thus the reader can look at the appendix whenever some notations are not clear. A more ample presentation of the partial algebraic framework can be found e.g. in [5,6,13].

1 Sound logical deduction for conditional specifications

In the following when referring to generic formulas and inference systems we consider formulas and inference systems within an infinitary logic which extends first-order logic by admitting countable conjunctions (, disjunctions) and quantification over countable sets of variables (see e.g.[9]). However we will show that we can restrict ourselves to consider only conditional formulas.
Let us recall in one definition some basic notions, just in order to fix the notation.

Def. 1.1. Let $\Sigma = (S,F)$ be a signature and X be a family of S-sorted variables.

- If $t,t' \in T_\Sigma(X)$, then $D(t)$ and $t = t'$ are *elementary formulas*.
  If Δ is a countable set of elementary formulas and ε is an elementary formula too, then $\wedge \Delta \supset \varepsilon$ is a *conditional formula.*
  If Δ is the empty set, then $\wedge \Delta \supset \varepsilon$ is an equivalent notation for the elementary formula ε. ($\wedge \Delta$ is a notation for the couple $(\wedge,\Delta)$; see [9]).
- A *positive conditional formula* is a conditional formula $\wedge \Delta \supset \varepsilon$ s.t. for every $t = t'$ belonging to Δ either $D(t)$ or $D(t')$ belongs to Δ.
- For every formula φ let $Var(\varphi)$ denote the set of all variables which appear in φ. A formula φ is called *closed* iff $Var(\varphi)$ is empty.
- If A is a partial algebra, φ is a formula and V is a valuation for $Var(\varphi)$ in A, then we say that φ holds for V in A (equivalently: is satisfied for V by A) and write $A \models_V \varphi$ accordingly to the following definitions
  - let φ be $D(t)$; then $A \models_V D(t)$ iff $t^{A,V}$ is defined; let φ be $t = t'$; then $A \models_V t = t'$ iff $t^{A,V}$ and $t'^{A,V}$ are either both defined and equal or both undefined;
  - let φ be $\wedge \Delta \supset \varepsilon$; then $A \models_V \varphi$ iff either $A \not\models_V \varepsilon$ or $A \models_V \delta$ for some $\delta \in \Delta$.

  We write $A \models \varphi$ for a formula φ and say that φ holds in (equivalently: is satisfied by, is valid in) A iff $A \models_V \varphi$ for all valuations V for $Var(\varphi)$ in A.
- A *(positive) conditional type (specification)* T consists of a signature Σ and of a set Ax of (positive) conditional formulas over Σ, the *axioms* of T. A generic conditional type will be denoted by T and a formula belonging to Ax by α.

- For every conditional type $T = (\Sigma, Ax)$, $PMod(T)$ denotes the class of all *models* of T, ie the Σ-algebras satisfying every formula of Ax;
  $PMod(T) = \{A \mid A \in PA(\Sigma),\ A \models \alpha,\ \forall\ \alpha \in Ax\}$. fi

In the following a generic elementary formula will be denoted by ε or γ or δ, while a generic conditional formula will be denoted by φ; moreover for all conditional formulas $\varphi = (\wedge\ \Delta \supset \varepsilon)$ we denote Δ by $prem(\varphi)$ and ε by $cons(\varphi)$; finally we use some equivalent notations:

- $\wedge\ \Delta_1 \wedge \ldots \wedge \wedge\ \Delta_n$ is the same as $\wedge\ (\cup_{i=1\ldots n}\ \Delta_i)$, where Δ_i is a countable set of elementary formulas for $i=1\ldots n$;
- $\varepsilon_1 \wedge \ldots \wedge \varepsilon_n$ is the same as $\wedge\{\varepsilon_1, \ldots, \varepsilon_n\}$, where ε_i is an elementary formula for $i=1\ldots n$;
- $D(X)$ is the same as $\{D(x) \mid x \in X\}$, where X is a countable family of variables.

Note that, as usual, quantification is always implicit and is universal, ie every formula φ is a short notation for the formula $\{\forall\ x: s \mid x \in Var(\varphi)_s\}_{s \in S} : \varphi$. However this short notation can induce in a subtle error whenever empty carriers are allowed; consider the following example.

Example 1: specification T_1

sorts: s_1, s_2 operations: $a,b: \rightarrow s_1$ axioms: $D(a), D(b)$

$f: s_2 \rightarrow s_1$ $a = f(x),\ f(x) = b$

Now if $L(T_1)$ is a logical system for T_1, then we obviously have $L(T_1) \models a = f(x)$ and $L(T_1) \models f(x) = b$, while $L(T_1) \models a = b$ is an unsound deduction, because T_Σ is the initial model of T_1 and $T_\Sigma \not\models a = b$. **End**

First Huet noted, in the framework of total many-sorted algebras, that the family

$\Re = \{(t,t') \mid t,t' \in T_\Sigma(X)_{|s}\ A \models t = t'\}_{s \in S}$

is not a congruence; for example in the specification T_1 both $T_\Sigma \models a = f(x)$ and $T_\Sigma \models f(x) = b$, because $T_{\Sigma|s_2} = \varnothing$ and hence there does not exist any valuation for $\{x\}$ in T_Σ, but $T_\Sigma \not\models a = b$ so that $\Re$ is not transitive. To avoid the problem he suggested to restrict signatures to the only ones whose carriers either are guaranteed to be non-empty by the existence of closed terms of that sort, or are in a sense absolutely disconnected by the non-empty carriers (the rigourous notion is that of sensible signature, see e.g. [8]). This approach fails in the partial framework since a closed term may be undefined in an algebra and hence its existence does not guarantee that the corresponding carrier is not empty; thus we cannot guarantee that all carriers are not empty just because of the signature.

The problem was also developed by [10] with a particular interest to logical deduction. In [10] the quantification is made explicit as usual in classical logic; thus the system works on equalities of the form $(\forall X)\ t = t'$, and produces $(\forall X\text{-}\{x\})\ t = t'$, eliminating a variable x from X, only if x does not appear in $t = t'$ and can be instantiated by a closed term. In that framework in example 1 we have that from $(\forall\{x\})\ f(x) = b$ and $(\forall\{x\})\ a = f(x)$ we deduce $(\forall\{x\})\ a = b$, which holds also in T_Σ, but we cannot deduce $a = b$, as x cannot be instantiated on a closed term, $T_{\Sigma|s_2}$ being empty. A similar

approach can be used also in the partial framework, eliminating variables that can be instantiated on closed terms whose definedness is provable.

However we can also handle the problem in a more economical way, explicitly using definedness predicates. In order to keep memory of extra-variables used in the deduction we can simply add $D(y)$ to the premises of the deduced formula for all "extra-variables" y; indeed for all variables y the only effect of the presence of $D(y)$ in the left-hand side of a conditional axiom is increasing the set of variables appearing in the axiom, since obviously the formula $D(y)$ holds in all algebras. Thus the [10]-like formula $(\forall X) \wedge \Delta \supset \varepsilon$ is completely equivalent to $\wedge D(X) \wedge \wedge \Delta \supset \varepsilon$.

Let us present now a system which takes care of the empty-carriers problem using the above remark. This system is reminiscent of systems found in the literature (see, e.g. [14]) and it is easy to check that it is complete for positive conditional types w.r.t. open *existential* equalities, but it is not complete both for (non-positive) conditional types w.r.t. *existential* equalities and for positive conditional types w.r.t. *strong* equalities. In the next section we will add to this system just one rule and obtain a system complete w.r.t. open strong equalities for all (non-positive) conditional types.

Def. 1.2. The CL(T) c-system for a conditional type $T = (\Sigma, Ax)$ consists of the axioms Ax and of the following axioms and inference rules:

0	$D(x)$	all variables x
1	$t = t$	all open terms t
2	$t = t' \supset t' = t$	all open terms t,t'
3	$t = t' \wedge t' = t'' \supset t = t''$	all open terms t,t',t''
4	$t_1 = t'_1 \wedge \ldots \wedge t_n = t'_n \supset op(t_1,\ldots,t_n) = op(t'_1,\ldots,t'_n)$	all open terms t_i, t'_i of sort s_i, $i=1\ldots n$, op: $s_1 \times \ldots \times s_n \to s$
5	$D(op(t_1,\ldots,t_n)) \supset D(t_i)$	all open terms t_i of sort s_i, $i=1\ldots n$, op: $s_1 \times \ldots \times s_n \to s$
6	$D(t) \wedge t = t' \supset D(t')$	all open terms t,t'
7	$\dfrac{\wedge \Delta \wedge \wedge \Gamma \supset \varepsilon, \ \{\wedge \Delta_\gamma \supset \gamma \mid \gamma \in \Gamma\}}{\wedge D(Var(\Gamma)\text{-}Var(\cup_{\gamma\in\Gamma} \Delta_\gamma)) \wedge \wedge \Delta \wedge \wedge (\cup_{\gamma\in\Gamma} \Delta_\gamma) \supset \varepsilon}$	$\Delta, \Delta_\gamma, \Gamma$ are arbitrary, countable sets of elementary formulas, ε is an elementary formula.
8	$\dfrac{\wedge \Delta \supset \varepsilon}{\wedge X_t \wedge \wedge \{\delta[\{t_x/x \mid x\in X_s, s\in S\}] \mid \delta\in\Delta\} \supset \varepsilon[\{t_x/x \mid x\in X_s, s\in S\}]}$	$X_t = \{D(t_x) \mid x\in X_s \ s\in S\}$, Δ is an arbitrary, countable set of elem. formulas, ε is an elem. formula, t_x are open terms of sort s for all $x\in X_s$.

Prop. 1.3. The system CL(T) is sound for all conditional specifications T. fi

2 A complete conditional system

In the previous section we have seen a sound logical system for conditional types. It is easy to show directly that if T is a positive conditional type, then $CL(T)$ is complete w.r.t. open existential equalities (for a similar prove see e.g. [10]). However if T is a non-positive conditional type, then $CL(T)$ may be not complete w.r.t. open existential equalities, as we show in the following example.

Example 2: specification T_2

sorts:	s_1, s_2	axioms:	α_1 $D(a) \supset D(e)$;
operations:	$a,b: \rightarrow s_1$		α_2 $a = b \supset D(e)$;
	$e: \rightarrow s_2$		α_3 $D(b) \supset D(e)$;

In all models of T_2, either $D(a)$ or $D(b)$ or $a = b$ holds, by definition of strong equality; thus, because of α_1, α_2 and α_3, also $D(e)$ holds in all models of T_2, while $D(e)$ cannot be deduced using only the rules of $CL(T_2)$. **End**

The example 2 suggests that, to make $CL(T)$ complete, we have to add a rule generalizing the one which sufficies for ground deduction and finitary axioms

$$\#\qquad \frac{\wedge(\Delta_1 \cup \{D(t)\}) \supset \varepsilon,\ \wedge(\Delta_2 \cup \{D(t')\}) \supset \varepsilon,\ \wedge(\Delta_3 \cup \{t = t'\}) \supset \varepsilon}{\wedge(\Delta_1 \cup \Delta_2 \cup \Delta_3) \supset \varepsilon}$$

where t and t' are closed terms. If t and t' are not closed, we have obviously to generalize # by keeping track of the variables in t and t' in the way introduced in sec.1.
However, since we are working within infinitary logic, we have to generalize # also to eliminate an infinite number of premises in one step.
Just in order to capture some intuitions about the needed generalization, let us first consider a finitary case where there are more then one strong equalities to eliminate (though every finitary case can be handled by a finite number of applications of #).

Example 3: specification T_3

sorts: s_1, s_2 operations: $a,b,c,d: \rightarrow s_1$; $e: \rightarrow s_2$

axioms:

α_1 $D(a) \wedge D(c) \supset D(e)$; α_2 $a = b \wedge D(c) \supset D(e)$; α_3 $D(b) \wedge D(c) \supset D(e)$;
α_4 $D(a) \wedge D(d) \supset D(e)$; α_5 $a = b \wedge D(d) \supset D(e)$; α_6 $D(b) \wedge D(d) \supset D(e)$;
α_7 $D(a) \wedge c=d \supset D(e)$; α_8 $a = b \wedge c=d \supset D(e)$; α_9 $D(b) \wedge c=d \supset D(e)$.

In all models of T_3, by definition of strong equality, at least one among $D(a)$, $D(b)$, $a = b$ and one among $D(c)$, $D(d)$, $c = d$ holds. Therefore in all models of T_3 the premises of at least one among $\alpha_1,\ldots,\alpha_9$ hold and hence we conclude that $D(e)$ holds in all models of T_5. **End**

Note that in all models of T_3 the premises of at least one axiom hold since $\{prem(\alpha_i) \mid i=1\ldots9\}$ is the set $\{D(a),D(b),a = b\}\times\{D(c),D(d),c = d\}$ and one among $\{D(a),D(b),a = b\}$ and one among $\{D(c),D(d),c = d\}$ has to hold. Then for a generic

finitary case we have that we deduce from a family $\{\varphi_i \mid i=1\ldots m\}$ of conditional formulas an elementary formula ε iff:

- $cons(\varphi_i) = \varepsilon$ for all $i=1\ldots m$;
- $\{prem(\varphi_i) \mid i=1\ldots m\} = \{D(t_1),D(t'_1),t_1 = t'_1\}\times\ldots\times\{D(t_n),D(t'_n),t_n = t'_n\}$ for suitable t_j,t'_j and $j=1\ldots n$.

Indeed in all models A one among $D(t_j)$, $D(t'_j)$, $t_j = t'_j$ holds for all $j=1\ldots n$ and hence there exists $i\in\{1\ldots m\}$ s.t. $A \models \delta$ for all $\delta\in prem(\varphi_i)$.

In order to generalize this notion to the infinitary case, let us first introduce a short notation.

If Γ_i is a set of elementary formulas for all $i\in I$, then $FullInter(\{\Gamma_i \mid i\in I\})$ denote the set of all $\Psi \subseteq \cup_{i\in I}\Gamma_i$ s.t. $\Psi \cap \Gamma_i \neq \emptyset$ for all $i\in I$.

If $\{\Gamma_i \mid i=1\ldots m\}$ is $\Pi = \{D(t_1),D(t'_1),t_1 = t'_1\}\times\ldots\times\{D(t_n),D(t'_n),t_n = t'_n\}$, then for all $\Psi \in FullInter(\{\Gamma_i \mid i=1\ldots m\})$ there exists $j\in\{1,\ldots,n\}$ s.t. $\{D(t_j),D(t'_j),t_j = t'_j\} \subseteq \Psi$; indeed if for all $j=1\ldots n$ there exists $\delta_j\in\{D(t_j),D(t'_j),t_j = t'_j\}$ s.t. $\delta_j\notin\Psi$, then $\Delta=(\delta_1,\ldots,\delta_n)\in\Pi$, but $\Delta\cap\Psi$ is empty.

Then the intuition of the needed generalization is

- for all $\Psi\in FullInter(\{prem(\varphi_i) \mid i\in I\})$ there exist $t,t'\in T_\Sigma$ s.t. $\{D(t),D(t'),t = t'\} \subseteq \Psi$.

Let us formalize this idea.

Def. 2.1. The *canonical* c-system for a conditional type T, denoted by $CL_v(T)$ where v stands for variables deduction, consists of the axioms and inference rules of $CL(T)$ and of the following rule:

9 $$\frac{\{\wedge\Delta_i \wedge \wedge\Gamma_i \supset \varepsilon \mid i\in I\}}{\wedge D(Z) \wedge \wedge (\cup_{i\in i}\Delta_I) \supset \varepsilon}$$

I is an arbitrary set (possibly more than countable),
Δ_i, Γ_i are arbitrary countable sets of elementary formulas,
ε is an elementary formula too,
$Z = ((\cup_{i\in I} Var(\Gamma_i))\text{-}Var(\wedge(\cup_{i\in I}\Delta_i) \supset \varepsilon))$.
$\forall\ \Psi\in FullInter(\Gamma)\ \exists$ terms t,t' s.t. $D(t),D(t'),t=t'\in\Psi$, where
$FullInter(\Gamma)=\{\Psi \mid \Psi\subseteq(\cup_{i\in I}\Gamma_i),\Psi\cap\Gamma_i\neq\emptyset,\forall\ i\in I\}$. fi

Prop. 2.2. For all conditional specifications T the system $CL_v(T)$ is sound. fi

It is easy to see that this system is a generalization of the one in [1], from now on denoted by $CL_g(T)$, where g stands for ground deduction, following the idea outlined above of adding to the premises of deduced formulas a set of the form $D(X)$, so that the variables of a formula only decrease by rule 7. Thus the completeness of $CL_v(T)$ can be deduced from the one of $CL_g(T)$.

Def. 2.3. Let T be a conditional specification and X be a family of variables. A deduction system $L(T)$ is *complete* for T w.r.t. X iff for any equalities $t = t'$ on X, if $M \models t = t'$ $\forall\ M\in PMod(T)$, then $L(T) \vdash \wedge D(X) \supset t = t'$.

Theorem 2.4. The system $CL_v(T)$ is complete for T w.r.t. all families X of variables.

Proof outline. The proof relies on a result of [1], which we briefly recall.

Let $CL_g(T)$ be the system defined in def.3.1 of [1], ie $CL_g(T)$ consists of the axioms of T, the rules 1…6,8,9 of $CL_v(T)$ for only closed terms and formulas and rule 7 of $CL_v(T)$, where instantiation only range on closed terms. Then theorem 3.7 of [1] says that $CL_g(T)$ is complete w.r.t. closed elementary formulas.

Now let us introduce some notations. Let Σ_X be $(S,F\cup\{\underline{x}: \rightarrow s \mid x\in X_s\}_{s\in S})$, ie Σ increased by a constant symbol $\underline{x}$ for all variables x, T_X be the conditional specification $(T_X,Ax\cup\{D(\underline{x})\}_{x\in X})$.

$CL_g(T_X) \vdash t[\underline{x}/x|x\in X] = t'[\underline{x}/x|x\in X]$ implies $CL_v(T) \vdash \wedge D(X) \supset t = t'$, as it can be easily shown by induction on $CL_g(T_X)$, for all equalities $t = t'$.

Let us assume that $CL_v(T) \nvdash \wedge D(X) \supset t = t'$ and show that there exist $B\in PMod(T)$ and a valuation V for X in B s.t. $B \nvDash_V t = t'$. Since we have assumed that $CL_v(T) \nvdash \wedge D(X) \supset t = t'$, $CL_g(T_X) \nvdash t[\underline{x}/x|x\in X] = t'[\underline{x}/x|x\in X]$ and hence, being $CL_g(T_X)$ complete, there exists a model A of T_X s.t. $A \nvDash t[\underline{x}/x|x\in X] = t'[\underline{x}/x|x\in X]$. Now note that if A is a model of T_X, then $A_{|\Sigma}$ (the reduct of A, see e.g. [6]) is a model of T and $U(x) = \underline{x}^A$ is a valuation for X in $A_{|\Sigma}$. Therefore $A_{|\Sigma}$ is a model of T and $A_{|\Sigma} \nvDash_U t = t'$. fi

3 Free objects and logical deduction

In this section we connect the non-existence of free objects with the deducibility of a certain kind of formulas, called "naughty", by the system $CL_v(T)$. More exactly we first show that (as usual both in partial positive and total conditional specifications) the quotient of $T_\Sigma(X)$ naturally associated with $CL_v(T)$, in the following denoted by Fr_v, is the free object for X in $PMod(T)$ iff there exists a free object for X in $PMod(T)$. Then we show that Fr_v is a model of T iff there does not exist any naughty formula.

Let us first introduce some notation.

Def. 3.1. For all conditional types T and all families X of variables
the congruence $\equiv(T,X)$ is the family
$\{(t,t') \mid t,t'\in T_\Sigma(X)_{|s}, CL_v(T) \vdash \wedge D(X) \supset D(t), CL_v(T) \vdash \wedge D(X) \supset t=t'\}_{s\in S}$;
the algebra $Fr(T,X)$ is $T_\Sigma(X)/\equiv(T,X)$.

In the following if there is not any ambiguity we will shortly denote $\equiv(CL_v(T),X)$ by $\equiv$, and $Fr(T,X)$ by Fr; moreover we will denote by m the valuation $m: X \rightarrow Fr$ defined by $m(x) = [x]$. fi

Remark. Note that, because of rules 1,…,7, the family $\equiv$ is really a congruence; moreover, because of rule 0, m is always a valuation for X in Fr.

Then we state the equivalence between Fr being the free object for X in $PMod(T)$ and the existence of a free model for X in $PMod(T)$.

Prop. 3.2. For all conditional specifications T and all families X of variables the following conditions are equivalent.

1) The algebra Fr is a model of T.
2) The couple (Fr,m) is free for X in $PMod(T)$.
3) There exists a free object for X in $PMod(T)$. fi

In both frameworks, partial positive conditional and total conditional specifications, the quotient of the term algebra w.r.t. the congruence naturally associated with a sound and equationally-complete logical system is a model of the specification. In the following example we show that such property does not hold for (non-positive) conditional specifications.

Example 4: Specification T_4

Sort: s operation symbols: a,b,c: $\rightarrow$ s axioms: $D(c) \wedge a = b \supset D(a)$
$D(c)$

Obviously the two algebras where c and just one between a and b are defined are models of T_4; thus, $CL_v(T_4)$ being sound, both $CL_v(T_4) \vdash D(a)$ and $CL_v(T_4) \vdash D(b)$.
Therefore $Fr_v \models_m a = b$ and, since $CL_v(T_4) \vdash D(c)$, $Fr_v \models_m D(c)$, while $Fr_v \not\models_m D(a)$ and hence Fr_v is not a model of T_4. **End**

Since in general Fr_v is not a model of T we have to look for necessary and sufficient conditions guaranteeing $Fr_v \in PMod(T)$.
Example 4 suggests that the problem arises because of some (non-open) axioms whose premises hold in Fr while the consequence does not. Consider another example where there are some non-closed axioms.

Example 5: Specification T_5

Sorts: s axioms: α_1 $D(a)$;
operation symbols: a,b: $\rightarrow$ s α_2 $b = f(x) \supset a = b$
f: s $\rightarrow$ s

Then by α_1 we have that $CL_v(T_5) \vdash D(a)$, so that, by instantiation of α_2, we also have that $CL_v(T_5) \vdash b = f(a) \supset a = b$; moreover, as it is easy to check, $CL_v(T_5) \nvdash D(f(a))$ and $CL_v(T_5) \nvdash D(b)$, so that $Fr \models_m b = f(a)$, while $Fr \not\models_m a = b$, since $Fr \models_m D(a)$ and $Fr \not\models_m D(b)$. **End**

Generalizing this idea we have that Fr is not a model, since there exists an instantiation of an axiom whose premises hold in Fr w.r.t. m and the consequence does not. This idea leads us to define the set of Naughty Formulas.

Def. 3.3. For all conditional types T and all families X of variables the set NF(T,X) consists of all conditional formulas φ s.t.

nf_1 φ is $\alpha[t_y/y \mid y \in Var(\alpha)]$ for some $\alpha \in Ax$ and $t_y \in T_\Sigma(X)$ s.t. $Fr \models_m D(t_y)$;
nf_2 $Fr \models_m \delta$ for all $\delta \in prem(\varphi)$;
nf_3 $Fr \not\models_m cons(\varphi)$. fi

From the definition of NF(T,X) it is easy to understand that $Fr \in PMod(T)$ iff $NF(T,X) = \emptyset$.

Theorem 3.4. For all conditional types T and all families X of variables $Fr \in PMod(T)$ iff $NF(T,X) = \emptyset$. fi

Let us collect now all the results about the existence of free objects.

Theorem 3.5. Let T be a conditional type and X be a family of variables. The following conditions are equivalent:

1) the set $NF(T,X)$ is empty;
2) the algebra Fr is a model of T;
3) the couple (Fr,m) is free for X in $PMod(T)$;
4) there exists a free object for X in $PMod(T)$. fi

Note that if α is a positive conditional axiom, then it is impossible that any of its instantiations belongs to $NF(T,X)$. Thus we can instantiate the theorem for positive conditional specifications.

Corollary 3.6 [6]**.** Let X be a family of variables and PT be a positive conditional type; then (Fr,m) is free for X in $PMod(PT)$. fi

Conclusions. Let us conclude with some hints for the first-order representation of higher-order types.

A higher-order specification FT on basic sorts S consists of a first-order specification $((S^{\to},F),Ax)$, where $S^{\to}$ is inductively defined by $S \subseteq S^{\to}$, $s_1,\dots,s_n,s \in S^{\to}$ implies $(s_1\times\dots\times s_n \to s)\in S^{\to}$, s.t.
Apply functions: $apply_s \in F_{(ss_1\dots s_n,s_{n+1})}$ for all $s = (s_1\times\dots\times s_n \to s_{n+1})\in S^{\to}$;
Term-extensionality: $[\wedge_{\underline{t}\in \underline{T}}\ apply_s(f,\underline{t}) = apply_s(g,\underline{t}) \supset f = g]\in Ax$,
for all $s = (s_1\times\dots\times s_n \to s_{n+1})\in S^{\to}$, $\underline{T} = T_{\Sigma|s_1}\times\dots\times T_{\Sigma|s_n}$, $f,g\in Var_s$.

Assume now that the only non-positive axioms of FT are the term-extensionality axioms. In this case in order to have the existence of the free object on X we have just to guarantee that any instantiation of some term-extensionality axioms is not a naughty formula, ie that either $CL_v(T)$ " $\wedge$ $D(X) \supset apply_s(f,\underline{t}) = apply_s(g,\underline{t})$ for all $\underline{t}\in \underline{T}$ or $CL_v(T)$ " $\wedge$ $D(X) \supset t_f = t_g$ for all $t_f,t_g \in T_{F\Sigma|s}$ s.t.
$[CL_v(T)$ " $\wedge$ $D(X) \supset D(apply_s(f,\underline{t}))$ iff $CL_v(T)$ " $D(apply_s(g,\underline{t}))]$ and
$[\wedge\ D(X) \supset D(apply_s(f,\underline{t}))$ implies
$CL_v(T)$ " $\wedge$ $D(X) \supset apply_s(f,\underline{t}) = apply_s(g,\underline{t})]$.

Note that this condition requires less equalities between undefined terms than the usual "total" reduction, where a special symbol $\perp$ is introduced to represent the "undefined" and *all* undefined terms are equated to $\perp$.

Acknowledgements. This paper grew out of some joint work with E. Astesiano, as a part of a doctoral dissertation, currently developed under his supervision. I wish to thank him for his constant encouragement and help.

References

[1] Astesiano, E.; Cerioli, M. "On the Existence of Initial Models for Partial (Higher-Order) Conditional Specifications", *Proc. TAPSOFT'89*, vol.1, Berlin, Springer Verlag, 1989 (Lecture Notes in Computer Science n. 351), pp. 74–88.

[2] Astesiano, E.; Cerioli, M. "Free objects and equational deduction for partial (higher-order) conditional specifications", (Technical report, October 1989).

[3] Astesiano, E.; Reggio, G. "SMoLCS-Driven Concurrent Calculi", *Proc. TAPSOFT'87*, vol.1, Berlin, Springer Verlag, 1987 (Lecture Notes in Computer Science n. 249), pp. 169–201.

[4] Astesiano, E.; Reggio, G. "An Outline of the SMoLCS Methodology", (invited paper) *Mathematical Models for the Semantics of Parallelism, Proc. Advanced School on Mathematical Models of Parallelism* (Venturini Zilli, M. ed.), Berlin, Springer Verlag, 1987 (Lecture Notes in Computer Science n. 280), pp. 81-113.

[5] Burmeister, P. *A Model Theoretic Oriented Approach to Partial Algebras*, Berlin, Akademie-Verlag, 1986, pp. 1-319.

[6] Broy, M.; Wirsing, M. "Partial abstract types", *Acta Informatica* 18 (1982), 47-64.

[7] Broy, M.; Wirsing, M. "On the algebraic specification of finitary infinite communicating sequential processes", *Proc. IFIP TC2 Working Conference on "Formal Description of Programming Concepts II"*, Garmisch 1982.

[8] Huet, G. and Oppen, D. Equations and Rewrite Rules: A Survey. In *Formal Language Theory: Perspectives and Open Problems*, R. Book., Ed., Academic Press, 1980.

[9] Keisler, H.J. *Model Theory for Infinitary Logic*, Amsterdam - London, North-Holland Publishing Company, 1971, pp. 1-208.

[10] Meseguer, J.; Goguen, J.A. "Initiality, Induction and Computability", *Algebraic Methods in Semantics*, Cambridge, edited by M.Nivat and J.Reynolds, Cambridge University Press, 1985, pp.459-540.

[11] Möller B., Tarlecki A., Wirsing M. "Algebraic Specification with Built-in Domain Constructions", *Proceeding of CAAP '88 (Nancy France, March 1988)*, edited by Dauchet M. and Nivat M., Berlin, Springer-Verlag, 1988, pp. 132-148.

[12] Reichel H. *Initial Computability, Algebraic Specifications, and Partial Algebra*s, Berlin (D.D.R.), Akademie-Verlag, 1986.

[13] Tarlecki A. "Quasi-varieties in Abstract Algebraic Institutions", Journal of Computer and System Science, n. 33 (1986), pp. 333 - 360.

[14] Wirsing, M.; Broy, M. *An analysis of semantic models for algebraic specifications*, International Summer School on Theoretical Foundations of Programming Methodology, Munich. Germany 28/7 - 9/8, 1981.

Appendix

Def. A.1. A *signature*, usually denoted by Σ, is a couple (S,F), where S is a (countable) *sort* set, and F is an $S^*\times S$-indexed family of *operation symbols* sets s.t. $F_{(w,s)}\cap F_{(w',s)} \neq \emptyset$ implies $w = w'$; if $op\in F_{(s_1\ldots s_n,s)}$, then we usually write $op: s_1\times\ldots\times s_n \to s$. fi

Notice that the requirement $F_{(w,s)}\cap F_{(w',s)} = \emptyset$ for all $w \neq w'$ is the less restrictive in order to have not any ambiguity about the terms interpretation.

Def. A.2.

- A *partial* Σ*-algebra* A, in the following simply called Σ-algebra or algebra, is a couple $(\{s^A\}_{s\in S},\{op^A\}_{op\in F})$, where s^A is a set, called the *carrier* of sort s, for all $s\in S$ and op^A is a partial function $op: s_1^A\times\ldots\times s_n^A \to s^A$ for all $op: s_1\times\ldots\times s_n \to s$. In particular if $n = 0$, then op^A is either an element of s^A or it is undefined.
- The term algebra $T_\Sigma(X)$ on a family X of variables is defined in the usual total way (see e.g. [10]).
- For all valuations for X in an algebra A the natural interpretation of $T_\Sigma(X)$ w.r.t. V, denoted by $t^{A,V}$ or simply t^A if $X = \emptyset$, is inductively defined by:
  $x^{A,V}=V(x)$; $op(t_1,\ldots t_n)^{A,V}=op^A(t_1^{A,V},\ldots,t_n^{A,V})$ $\forall$ $op:s_1\times\ldots\times s_n\to s, t_i\in T_\Sigma(X)_{|si}$ $i=1\ldots n$.
- The *Kernel* of the natural interpretation of $T_\Sigma(X)$ w.r.t. a valuation V, denoted by $K^{A,V}$ or just K^A if $X=\emptyset$, is the congruence $\{(t,t') \mid t,t'\in T_\Sigma(X)_{|s},t^{A,V},t'^{A,V}\in s^A,t^{A,V} = t'^{A,V}\}_{s\in S}$.
- A *homomorphism* between two partial Σ-algebras A and B, denoted by $h: A \to B$, is a family $\{h_s\}_{s\in S}$ of total function $h_s: s^A \to s^B$ s.t. if $op^A(a_1,\ldots,a_n)\in s^A$, then $h_s(op^A(a_1,\ldots,a_n)) = op^B(h_{s_1}(a_1),\ldots h_{s_n}(a_n))$ for all $op: s_1\times\ldots\times s_n\to s$, $a_i\in s_i^A$ $i=1\ldots n$. fi

Notice that, being op^A a partial function, also the natural interpretation is partial and hence t^A may be undefined.

Def A.3. For all non empty classes C of partial Σ-algebras and all families X of variables a couple (Fr,f), where Fr is an algebra of C and f is a valuation for X in Fr, is *free* for X in C iff for all $A\in C$ and all valuations V for X in A there exists a unique homomorphism $h: Fr \to A$ s.t. $h(f(x)) = V(x)$ for all $x\in X$; if X is the empty set, then f has to be the empty map, so that the definition can be simplified as follows: I is initial in C iff for all $A\in C$ there exists a unique homomorphism $h: I \to A$. fi

Prop. A.4. Let C be a non-empty class of Σ-algebras.

1 If (Fr,f) is free for X in C, then $K^{Fr,f} = \cap_{A\in C,V:X\to A} K^{A,V}$; ie
   - $t^{Fr,f}\in s^{Fr}$ iff $(t^{A,V}\in s^A$ for all $A\in C$ and all $V:X\to A)$ for all $t\in T_\Sigma(X)$;
   - if $t^{Fr,f},t'^{Fr,f}\in s^{Fr}$, then $t^{Fr,f} = t'^{Fr,f}$ iff $(t^{A,V} = t'^{A,V}$ for all $A\in C$ and all $V:X\to A)$ for all $t,t'\in T_\Sigma(X)$.

2 If there exists $Fr\in C$ s.t. $\varphi: T_\Sigma(X)/\cap_{A\in C,V:X\to A} K^{A,V} \to Fr$ is an isomorphism, then (Fr,f), where $f(x) = \varphi([x])$ for all $x\in X$, is free for X in C.

3 If C is closed under isomorphisms and subobjects, then (Fr,m), where Fr is the algebra $T_\Sigma(X)/\cap_{A\in C,V:X\to A} K^{A,V}$ and $m(x) = [x]$ for all $x\in X$, is the free object for X in C iff there exists a free object for X in C, iff Fr belongs to C. fi