Relationships between Logical Frameworks*

Egidio Astesiano and Maura Cerioli[1]

DISI–Dipartimento di Informatica e Scienze dell'Informazione,
Università di Genova, Viale Benedetto XV, 16132 Genova, Italy,
e-mail: {astes,cerioli}@cisi.unige.it

Abstract. Adopting the concept of institution to represent logical frames, we have introduced in a previous paper the concept of simulation of an institution by another. Here we first show how simulations can be used to investigate the relationships between frames, distinguishing three levels, corresponding to different kinds of simulations: "set-theoretic", where the individual models of different frames are related disregarding their categorical and logical interconnection, "categorical", where the relation is between the categories of models, and "logical", where the relation is between specifications. Then we propose a concept of translation of inference systems along simulations such that soundness and completeness are preserved.

1 Introduction

Starting from the point of view that results should depend only on the very nature of the problems and not on the frame used to formalize them, in [2] we have introduced the concept of simulation of a logical formalism, viewed as an institution [8, 9], by another one and investigated the modularity properties of simulations w.r.t. specification languages and their relationship with the implementation relation.

The basic idea of simulation is encoding the syntax, i.e. signatures and sentences, of a new frame by that of an already known formalism in a way consistent with the semantics, in order to transfer results and tools. To formalize the consistency of the translation of the syntax w.r.t. the semantics, we require that every model of the new frame is represented by (at least) one of the old frame that satisfies the same sentences (under translation). Thus a simulation consists of three components: the two maps translating new signatures and sentences into old ones, and a partial surjective map which translates old models into new ones.

In this paper we use the concept of simulation mainly to analyse the different levels of relationship between frames and to translate inference systems from one institution to another, preserving soundness and completeness.

* This work has been partially supported by Esprit-BRA W.G. n.3264 Compass, Progetto Finalizzato Sistemi Informatici e Calcolo Parallelo of C.N.R. (Italy), MURST-40% Modelli e Specifiche di Sistemi Concorrenti

In the literature it is often claimed that a frame is *equivalent* to another one, usually in the sense that both solve the same kind of problems, or that in both the results are equivalently (un)satisfactory. But the meaning of equivalence is usually not formally defined and quite often used to denote different levels of relationship. Indeed we can distinguish (and formalize by means of simulations) three different levels, depending on whether the correspondence is between models, or categories of models, or specifications (theories). At the *set-theoretic* level, for every model in the new frame a model in the old frame can be found that represents the given one, in the sense that it satisfies the same formulas, or, more precisely, that it satisfies corresponding formulas. This is formalized by requiring that there exists a simulation from the new into the old frame (s.t. the domains of the model component corresponds to a, possibly non-full, subcategory of the old models). At this level most properties are missing, in particular no structured way of defining models is guaranteed to be preserved, because it usually involves categorical constructions. To have a *categorical* correspondence between two frames, at least the domain of the simulation has to be a full subcategory of the old models; moreover some more properties have to be required depending on the categorical structures that are intended to be preserved. Here we are focusing on the initial structures and give minimal conditions to preserve initiality; the analysis of the properties needed to deal with other categorical structures are still in progress. Even if there is a categorical simulation, the power of the specification languages in the two frames can be quite different; in particular it is possible that in the new frame some categories are definable by sets of sentences that are not so in the old one (and vice versa). To guarantee that the relationship is at the *logical level*, i.e. for every specification (i.e. the class of models which satisfy a set of sentences) in the new frame there exists a specification in the old frame equivalent to the given one in the categorical sense, we have to require not only that the domain of the model component is a full subcategory of the category of old models, but also that it is described by a set of old sentences.

The second point discussed in this paper is the translation of inference systems along simulations. Since every simulation can be seen as a coding of a new formalism into a known one, we are mostly interested in mapping inference systems from the old into the new institution. The basic idea for translating an inference system $\vdash'$ in the old institution $\mathcal{I}'$ along a simulation μ of $\mathcal{I}$ by $\mathcal{I}'$ is that an $\mathcal{I}$-formula ϕ is deducible from a set Φ of $\mathcal{I}$-formulas iff $\mu(\Phi) \vdash' \mu(\phi)$. In this way we build an inference system for $\mathcal{I}$ which consists of a preprocessing (the coding of both the premises and the consequence in terms of $\mathcal{I}'$-sentences), the running of the system $\vdash'$ and possibly a post-processing (the decoding of the answer). Since the validity of sentences is preserved by simulation and every $\mathcal{I}$-model is represented by at least one $\mathcal{I}'$-model, the soundness of $\vdash'$ w.r.t. $\mathcal{I}'$ guarantees the soundness w.r.t. $\mathcal{I}$ of its translation. Moreover, since every $\mathcal{I}$-model is simulated by some $\mathcal{I}'$-model, if $\vdash'$ is complete w.r.t. a class Φ of $\mathcal{I}'$-sentences and the domain of the simulation, then $\vdash'$ is complete w.r.t. every class of $\mathcal{I}$-sentences whose image along the simulation is contained in Φ and the $\mathcal{I}$-models. Thus simulations reflect sound and complete inference systems. As an application and an illustration of the above results we show that a sound and equationally complete inference system for the partial conditional higher-order frame can be obtained starting from every sound and equationally complete inference system for the partial conditional first-order frame.

The paper is organized as follows. In section 2 the basic definitions of institution and simulation are presented with the help of a simple example; then simulations are used to formally define three different levels of relationship between institutions and illustrated by the hierarchy of encoding partiality in the total frame. Section 3 is devoted to the translation of inference systems along simulation, to the proof that soundness and completeness are preserved and to the instantiation of these results to two basic examples. Finally, in section 4 some related work is mentioned and compared to ours.

2 Simulations and Relationships between Logical Frames

In this section we illustrate by a case study the different levels of relationship between logical frames and how this difference is captured by the notion of simulation.

2.1 Basic Simulations

To introduce the concept of simulation (see e.g. [2]) and the correspondent notation from an intuitive point of view, we begin with an informal example, which is the reduction of many-sorted equational Horn-clause logic, from now on $\mathcal{MS}$, to one-sorted Horn-clause logic, from now on $\mathcal{L}$, making explicit the typing of the variables (see e.g. [18]). In this example, as in the following ones, the notation for many-sorted open formulas has been slightly changed w.r.t. the usual algebraic notation, according to [9]. Indeed in order to make the translation of formulas along signature morphisms easier, a partial function $V\colon X \to S$, the *typing of variables*, is prefixed to any conditional formula on the signature (S, F) and variables $\{X_s\}_{s\in S}$, where $X_s = V^{-1}(s)$ are the s-typed variables. From now on we will assume that the domain of V is finitary for every formula $V.\phi$ in order to our translation work; to allow infinitary quantification in the many-sorted case, infinitary conjunctions in the premises are needed in the one-sorted case. In the following a simulation is denoted by μ, possibly decorated.

Example 2.1 Let us fix a many-sorted signature Σ with sorts S and function symbols F. We define the translation of Σ into a one-sorted signature $\mu_{Sign}(\Sigma)$, by setting $\mu_{Sign}(\Sigma) = (Op', P')$, where Op'_n is the disjoint union of $F_{s_1 \ldots s_n, s}$, i.e. of the n-ary function symbol sets, disregarding type of arguments and results (so that any Σ-term is an (Op', P')-term, too), and P' contains only the typing predicates, i.e. $P'_1 = \{\_ : s \mid s \in S\}$, where the symbol $\_$ denotes the place of the argument in a postfix notation, and $P'_k = \emptyset \quad \forall k \neq 1$.

With the help of the typing predicates, any many-sorted conditional equation over Σ can be translated into a one-sorted equivalent one over $\mu_{Sign}(\Sigma)$; indeed let us consider a many-sorted formula $\xi = (V.t_1 = t'_1 \wedge \ldots \wedge t_n = t'_n \supset t = t')$ over Σ and the variables x_i, where $V(x_i) = s_i$ for $i = 1 \ldots k$, and define

$$\mu_{Sen\,\Sigma}(\xi) = (x_1 : s_1 \wedge \ldots \wedge x_k : s_k \wedge t_1 = t'_1 \wedge \ldots \wedge t_n = t'_n \supset t = t').$$

Then in $\mu_{Sen\,\Sigma}(\xi)$, the translation of ξ over μ_{Sign}, the information about the typing of the variables is carried by the predicates $x_i : s_i$ in the premises.

To illustrate in which sense $\mu_{Sen\,\Sigma}(\xi)$ is equivalent to ξ, a class $dom(\mu)_\Sigma$ of one-sorted algebras is chosen, which soundly represents the many-sorted algebras and s.t. a one-sorted algebra satisfies $\mu_{Sen\,\Sigma}(\xi)$ iff the many-sorted algebra represented by it satisfies ξ. Again the typing predicates are used to simulate the different carriers of a many-sorted algebra: a one-sorted algebra $\mathtt{A}'$ is a sound representation of a many-sorted algebra A, we write $\mathsf{A} = \mu_{Mod\,\Sigma}(\mathtt{A}')$, iff whenever the arguments of a function are appropriately typed also the result is appropriately typed, i.e. $a_i : s_i^{\mathtt{A}'}$ for $i = 1 \ldots n$ implies $f^{\mathtt{A}'}(a_1, \ldots, a_n) : s^{\mathtt{A}'}$ for any $f \in F_{s_1 \ldots s_n, s}$. If $\mathtt{A}'$ satisfies this condition, then A is the many-sorted algebra $(\{s^{\mathsf{A}}\}_{s \in S}, \{f^{\mathsf{A}}\}_{f \in F})$, where $s^{\mathsf{A}} = \{a \mid a : s^{\mathtt{A}'}\}$ and f^{A} is the restriction of $f^{\mathtt{A}'}$ to $s_1^{\mathsf{A}} \times \ldots \times s_n^{\mathsf{A}}$; the above condition guarantees that the interpretation of the function symbols in A yields total functions. It is easy to check that $\mathtt{A}'$ satisfies $\mu_{Sen\,\Sigma}(\xi)$ iff A satisfies ξ. Note that one-sorted algebras differing only on elements which do not satisfy any typing predicate represent the same many-sorted algebra.

Thus for every many-sorted signature Σ, a homogeneous signature $\mu_{Sign}(\Sigma)$ and two functions are defined: $\mu_{Sen\,\Sigma}$, which translates many sorted equational conditional sentences on Σ into homogeneous conditional sentences on $\mu_{Sign}(\Sigma)$ built on typing predicates and equalities, and $\mu_{Mod\,\Sigma}$, which partially translates homogeneous first-order structures on $\mu_{Sign}(\Sigma)$ into many-sorted algebras on Σ and is surjective, as it is immediate to check.

Since the change of notation, via signature morphisms, has a great relevance in the algebraic approach, being used for example to bind the actual to the formal parameters in parameterized specifications and to "put theories together to make specifications", we have to investigate the compatibility between the coding functions $\mu_{Sen\,\Sigma}$ and $\mu_{Mod\,\Sigma}$ defined for any signature Σ and the changes of notation.

Let $\bar{\sigma}: \Sigma_1 \rightarrow \Sigma_2$ be a morphism of many-sorted signatures, i.e. a pair of functions $\sigma: S_1 \rightarrow S_2$, renaming the sorts, and $\phi: F_1 \rightarrow F_2$ translating function symbols in a consistent way w.r.t. the sort renaming (i.e. if $f: s_1 \times \ldots \times s_n \rightarrow s$, then $\phi(f): \sigma(s_1) \times \ldots \times \sigma(s_n) \rightarrow \sigma(s)$). Then $\bar{\sigma}$ naturally induces a homogeneous signature morphism $\mu_{Sign}(\bar{\sigma}) = (\psi', \pi')$ from $\mu_{Sign}(\Sigma_1)$ into $\mu_{Sign}(\Sigma_2)$, defined by $\psi'(f) = \phi(f)$ for any $f \in F$ and $\pi'(\_ : s) = \_ : \sigma(s)$ for any $s \in S$. It is easy to check that the translation of sentences is compatible with signature morphisms, i.e. that $\mu_{Sen\,\Sigma_2}(\bar{\sigma}(\xi)) = \mu_{Sign}(\bar{\sigma})(\mu_{Sen\,\Sigma_1}(\xi))$, where the application of a signature morphism to a sentence is the usual renaming of function (and predicate) symbols, plus the obvious translation of variable typing in ξ. Instead the partiality of the translation of algebras makes the compatibility between the algebra translations and signature morphisms delicate. Indeed it is intuitive to expect that the translation along a signature morphism of a one-sorted algebra simulating a many-sorted algebra simulates the translation of that many-sorted algebra; more formally, recalling that algebras are translated along signature morphisms in a countervariant direction into their *reduct*, we have that if $\mathtt{A}' \in dom(\mu)_{\Sigma_2}$, then $\mathtt{A}'|_{\mu_{Sign}(\bar{\sigma})} \in dom(\mu)_{\Sigma_1}$ and $(\mu_{Mod_{\Sigma_2}}(\mathtt{A}'))_{|\bar{\sigma}} = \mu_{Mod_{\Sigma_1}}(\mathtt{A}'|_{\mu_{Sign}(\bar{\sigma})})$. But the converse of the first implication does not hold, i.e. $\mathtt{A}'|_{\mu_{Sign}(\bar{\sigma})} \in dom(\mu)_{\Sigma_1}$ does not imply $\mathtt{A}' \in dom(\mu)_{\Sigma_2}$, as illustrated by the following example.

Let Σ_2 be the many-sorted signature $(\{nat\}, (\{0\colon \to nat, inc, dec\colon nat \to nat\})$, Σ_1 be its subsignature $(\{nat\}, (\{0\colon \to nat, inc\colon nat \to nat\})$, and $\bar{\sigma}$ be the embedding of Σ_1 into Σ_2. Consider now the one sorted algebra $\mathtt{A}'$ on $\mu_{Sign}(\Sigma_2)$, defined by

$$
\begin{array}{lll}
|\mathtt{A}'| = \mathbb{Z} & & a : nat^{\mathtt{A}'} \iff a \in \mathbb{N} \\
0^{\mathtt{A}'} = 0 & inc^{\mathtt{A}'}(x) = x + 1 & dec^{\mathtt{A}'}(x) = x - 1
\end{array}
$$

Then $\mathtt{A}' \notin dom(\mu)_{\Sigma_2}$, because $0 : nat$ holds, but $dec^{\mathtt{A}'}(0) : nat$ does not and hence $dec^{\mathtt{A}'}$ on appropriately typed input yields an untyped output. However $\mathtt{A}'|_{\bar{\sigma}}$ is the same as $\mathtt{A}'$ but *dec* has been dropped, hence it obviously belongs to $dom(\mu)_{\Sigma_1}$.

Therefore we have a weaker condition (called *partial naturality*) for algebras than the one for sentences: if $\mathtt{A}' \in dom(\mu)_{\Sigma_2}$, then $\mathtt{A}'|_{\mu_{Sign}(\bar{\sigma})} \in dom(\mu)_{\Sigma_1}$ and $(\mu_{Mod_{\Sigma_2}}(\mathtt{A}'))_{|\bar{\sigma}} = \mu_{Mod_{\Sigma_1}}(\mathtt{A}'|_{\mu_{Sign}(\bar{\sigma})})$. □

Let us abstract from the above construction the general aspects of the coding of a *new* (many-sorted) into an *old* (one-sorted) formalism:

- to each *new* signature an *old* signature corresponds;
- to each *new* sentence an *old* sentence corresponds;
- not any *old* algebra represents a *new* one, but to each *new* algebra at least one *old* corresponds, so that *old* algebras are (partially) translated by a surjective mapping.

This scheme generalizes to the frame of institutions by lifting maps to the proper categorical objects, taking care of the delicate points due to the partiality of model translation, and requiring that the only non-categorical structure, i.e. the validity relation, is preserved by them.

Def. 2.2 [[8] def.14] An *institution* $\mathcal{I}$ consists of

- a category **Sign** of *signatures*;
- a functor $Sen\colon \mathbf{Sign} \to \mathbf{Set}$ giving the set of *sentences* over a given signature;
- a functor $Mod\colon \mathbf{Sign}^{op} \to \mathbf{Cat}$[2] giving the category of *models* of a given signature;

[2] Usually **Cat** denotes the category of small categories; but in most significant examples from computer science non-small categories are needed as models and hence we use **Cat** to denote the category of all the categories whose objects belong to a suitable universe, that we never mention, as usual. In this way we avoid the well known foundational problems arising whenever one speaks of the *category of all the categories*. For a similar remark see also [9, 13, 22]

– a satisfaction relation $\models \subseteq |Mod(\Sigma)| \times Sen(\Sigma)$[3] for each Σ in **Sign**, sometimes denoted $\models_\Sigma$, such that for each morphism $\phi: \Sigma_1 \to \Sigma_2$ in **Sign**, the *Satisfaction Condition*

$$M' \models Sen(\phi)(\xi) \iff Mod(\phi)(M') \models \xi$$

holds for each M' in $|Mod(\Sigma_2)|$ and each ξ in $Sen(\Sigma_1)$. □

Since models are partially mapped, the usual notion of natural transformation is insufficient to describe the translation of the (*old*) model functor and we have to explicitly deal with the *partiality* of each component of this "partially"-natural transformation.

Def. 2.3 Let $\mathcal{I} = (\mathbf{Sign}, Sen, Mod, \models)$ and $\mathcal{I}' = (\mathbf{Sign}', Sen', Mod', \models')$ be institutions. Then a *simulation* $\mu: \mathcal{I} \to \mathcal{I}'$ consists of

– a functor $\mu_{Sign}: \mathbf{Sign} \to \mathbf{Sign}'$;
– a natural transformation $\mu_{Sen}: Sen \to Sen' \cdot \mu_{Sign}$, i.e. a natural family of functions $\mu_{Sen_\Sigma}: Sen(\Sigma) \to Sen'(\mu_{Sign}(\Sigma))$, and
– a surjective *partially-natural* transformation $\mu_{Mod}: Mod' \cdot \mu_{Sign} \to Mod$, that is a family of functors $\mu_{Mod_\Sigma}: dom(\mu)_\Sigma \to Mod(\Sigma)$, where $dom(\mu)_\Sigma$ is a (non-necessarily full) subcategory of $Mod'(\mu_{Sign}(\Sigma))$ s.t.
  - μ_{Mod_Σ} is surjective on $|Mod(\Sigma)|$;
  - the family is partially-natural, i.e. for any signature morphism $\sigma \in \mathbf{Sign}(\Sigma_1, \Sigma_2)$

$$Mod(\sigma) \cdot \mu_{Mod_{\Sigma_2}} = [\mu_{Mod_{\Sigma_1}} \cdot Mod'(\mu_{Sign}(\sigma))]_{|dom(\mu)_{\Sigma_2}}$$

s.t. the following *satisfaction condition* holds:

$$A \models \mu_{Sen_\Sigma}(\xi) \iff \mu_{Mod_\Sigma}(A) \models \xi$$

for all $\Sigma \in |\mathbf{Sign}|$, all $A \in |dom(\mu)_\Sigma|$ and all $\xi \in Sen(\Sigma)$. □

Note that the partial-naturalitycondition implies the following condition

$$Mod'(\mu_{Sign}(\sigma))(dom(\mu)_{\Sigma_2}) \subseteq dom(\mu)_{\Sigma_1}.$$

In the sequel, for any simulation μ, we will use μ also to denote its components, if the context makes clear the nature of the component.

It is easy to check that $\mu: \mathcal{MS} \to \mathcal{L}$, whose components were informally sketched in Example 2.1, is a simulation, from now on denoted μ^M (the superscript M stands for *Many-sorted*) in order to reserve the symbol μ to denote a generic simulation.

[3] for any category $\mathbf{C}$ we denote by $|\mathbf{C}|$ the class of the objects of $\mathbf{C}$.

2.2 A Paradigmatic Example: Partial versus Total Specifications

Here we illustrate the use of the notion of simulation, with its various specializations, as a tool for understanding the relationship between two formalisms with respect to the solution of a problem. We have chosen as a paradigmatic example the specification of (strict) partial functions in a partial and a total frame. The following analysis, though not pretending to be exhaustive especially on the pragmatic side, will highlight the subtleties of the relationship between the two frames and possibly reveal some misbeliefs.

Semantic Level. First we analyse the relationship between partial and total frames from a semantic point of view, i.e. disregarding their logics. Formally this means that we are working on institutions *without sentences*. Let us recall the basic ingredients of the partial frame (see [6, 7, 20]).

Def. 2.4 A *partial algebra* A on a signature $\Sigma = (S, F)$ consists of a family $\{s^A\}_{s \in S}$ of sets and of a family $\{f^A\}_{f \in F_{w,s}}$ of partial functions s.t. if $w = \emptyset$, then either f^A is undefined or $f^A \in s^A$, else $w = s_1 \dots s_n$ and $f^A : s_1^A \times \dots \times s_n^A \rightarrow_p s^A$. A *homomorphism* from A into B is a family $h = \{h_s : s^A \rightarrow s^B\}_{s \in S}$ of total functions, s.t. for any $f \in F_{s_1 \dots s_n, s}$ and any $a_i \in s_i^A$, for $i = 1 \dots n$, $f^A(a_1, \dots, a_n) \in s^A$ implies $h_s(f^A(a_1, \dots, a_n)) = f^B(h_{s_1}(a_1), \dots, h_{s_n}(a_n))$.

Let us denote by $\mathcal{PAR}_0$ the institution of partial algebras without sentences and by $\mathcal{MS}_0$ the institution of total many-sorted algebras without sentences. □

In the algebraic community there is a widespread belief that partiality can also be handled without explicit partial functions, in the usual total frame, simply by introducing a distinguished constant $\perp$ (one for each sort) to represent the undefined computations; in this way to any partial algebra A its trivial totalization corresponds. Following this intuition it is possible to define a *simulation* $\mu_0^\perp$ of partial by total algebras, where every partial algebra is simulated by its trivial totalization; but some homomorphisms between the trivial totalizations of partial algebras cannot be translated into homomorphisms of partial algebras, because the image of some *defined* element (i.e. of elements different from $\perp$) may be *undefined* (i.e. equal to $\perp$), while the homomorphisms of partial algebras are *total* functions. Therefore the domain of the simulation is not a full subcategory of the models and hence most categorical properties are missing. Let us formalize this in terms of simulation.

Def. 2.5 The simulation $\mu_0^\perp : \mathcal{PAR}_0 \rightarrow \mathcal{MS}_0$ consists of:

- $\mu_0^\perp : \mathbf{Sign}_{\mathcal{PAR}} \rightarrow \mathbf{Sign}_{\mathcal{MS}}$ is defined by $\mu_0^\perp((S, F)) = (S, F')$, where if $w \neq \emptyset$, then $F'_{w,s} = F_{w,s}$ else $F'_{\emptyset,s} = F_{\emptyset,s} \cup \{\perp_s\}$, and by $\mu_0^\perp((\sigma, \phi)) = (\sigma, \phi')$,where $\phi'(f) = \phi(f)$ for any $f \in F_{w,s}$ and $\phi(\perp_s) = \perp_{\sigma(s)}$.
- $\mu_0^\perp : \emptyset \rightarrow \emptyset$ is the empty natural transformation;
- $\mu_0^\perp : Mod_{\mathcal{MS}} \cdot \mu_0^\perp \rightarrow Mod_{\mathcal{PAR}}$ is defined by:

- $dom(\mu_0^\perp)_\Sigma$ is the subcategory of $Mod_{\mathcal{MS}}(\mu_0^\perp(\Sigma))$ whose objects are the total algebras A' s.t. $f^{\mathsf{A}'}(a_1,\ldots,a_n) \neq \perp_s$ implies $a_i \neq \perp^{\mathsf{A}'}{}_{s_i}$ for all $i = 1\ldots n$ for any $f \in F_{s_1\ldots s_n,s}$ (*strictness*) and whose arrows are the homomorphisms $\mathsf{h}' \in Mod_{\mathcal{MS}}(\mu_0^\perp(\Sigma))(\mathsf{A}',\mathsf{B}')$ s.t. $a \neq \perp_s^{\mathsf{A}'}$ implies $\mathsf{h}'_s(a) \neq \perp_s^{\mathsf{A}'}$ for any $s \in S$.
- for any $\mathsf{A}' \in dom(\mu_0^\perp)_\Sigma$ the partial algebra $A = \mu_0^\perp{}_\Sigma(\mathsf{A}')$ consists of $s^A = s^{\mathsf{A}'} - \{\perp_s^{\mathsf{A}'}\}$ for any $s \in S$ and for any $f \in F_{s_1\ldots s_n,s}$ and every $(a_1,\ldots,a_n)$, with $a_i \in s_i^A$ for $i = 1\ldots n$, if $f^{\mathsf{A}'}(a_1,\ldots,a_n) \neq \perp_s^{\mathsf{A}'}$, then $f^A(a_1,\ldots,a_n) = f^{\mathsf{A}'}(a_1,\ldots,a_n)$, else $f^{\mathsf{A}'}(a_1,\ldots,a_n)$ is undefined;
- $\mu_0^\perp{}_\Sigma(\mathsf{h}')$ is the restriction of h' to $s^{\mu_0^\perp{}_\Sigma(\mathsf{A}')}$. □

Note that formally strictness is not needed, because there are no sentences whose validity has to be preserved; however we prefer to require the strictness condition, because it is more intuitive specifying strict partial algebras and it will be needed in the sequel, to deal with logics.

Although obviously A' and $\mu_0^\perp{}_\Sigma(\mathsf{A}')$ are strictly related from a set theoretic point of view, the correspondence, due to the domain of $\mu_0^\perp$ being a non-full subcategory, is not adequate for categorical purposes, in particular the initial model is not preserved by $\mu_0^\perp$. Indeed in both frames the initial model is characterized by the well known *no junk* and *no confusion* conditions of [14], which mean that every element is denoted by some term and that two ground terms are equal in the initial object iff they are equal in every algebra of the class (in the partial frame the existential equality is considered, holding if both sides denote the same element of the carrier, so that also the *minimal definedness* holds). Thus the minimal equality (no-confusion) of the initial model in the total frame implies, in particular, the minimal equality with $\perp$ and hence the *maximal* definedness of its translation; therefore in most cases the translation of the initial model is not initial.

Categorical Level. We are now looking for simulations preserving properties like initiality. For this note that another way of coding partiality in terms of total algebras is to split every carrier by a typing predicate in typed (i.e. defined) and untyped elements and to represent every partial function by a total one which results in an untyped element over every input outside its domain (for similar approaches see e.g. [11], where one-sorted total algebras are used, [16] and [17]). Moreover, in order to handle logical formulas, in the following we also introduce a binary predicate, which plays the role of the existential equality, and holds on a' and b' iff a' and b' are equal and appropriately typed. The corresponding simulation is as follows.

Def. 2.6 Let us denote by $\mathcal{TL}_0$ the institution of typed first-order structures (total many-sorted algebras with predicates) without sentences.

The simulation $\mu^P{}_0: \mathcal{PAR}_0 \to \mathcal{TL}_0$ consists of:

- $\mu^P{}_0: \mathbf{Sign}_{\mathcal{PAR}} \to \mathbf{Sign}_{\mathcal{TL}}$ consists of $\mu^P{}_0((S,F)) = (S',F',P')$, where $S' = S$, $F' = F$ and if $w = ss$, then $P'_{ss} = \{eq_s\}$, if $w = s$, then $P'_s = \{D_s\}$, otherwise $P'_w = \emptyset$, and $\mu^P{}_0((\sigma, \phi)) = (\sigma', \phi', \pi')$, where $\sigma' = \sigma$, $\phi' = \phi$, $\pi'(D_s) = D_{\sigma(s)}$ and $\pi'(eq_s) = eq_{\sigma(s)}$.

– $\mu^P{}_0$ is the empty natural transformation;

– $\mu^P{}_0 \colon Mod_{\mathcal{TL}} \cdot \mu^P{}_0 \to Mod_{\mathcal{PAR}}$ is defined by:
  - $dom(\mu^P{}_0)_\Sigma$ is the full subcategory of $Mod_{\mathcal{TL}}(\mu^P(\Sigma))$ whose objects are the total algebras A' s.t. for any $f \in F_{s_1 \ldots s_n, s}$ if $D_s{}^{\mathsf{A}'}(f^{\mathsf{A}'}(a_1, \ldots, a_n))$ holds, then $D_{s_i}{}^{\mathsf{A}'}(a_i)$ holds, too, for all $i = 1 \ldots n$ (*strictness*) and $eq_s^{\mathsf{A}'}(a, b)$ iff $a = b$ and $D_s{}^{\mathsf{A}'}(a)$, $D_s{}^{\mathsf{A}'}(b)$.
  - for every $\mathsf{A}' \in dom(\mu^P)_\Sigma$ the partial algebra $A = \mu^P{}_{0\Sigma}(\mathsf{A}')$ consists of $s^A = D_s{}^{\mathsf{A}'}$ for any $s \in S$ and for any $f \in F$ and every $a_i \in s_i^A$ if $D_s{}^{\mathsf{A}'}(f^{\mathsf{A}'}(a_1, \ldots, a_n))$ holds, then $f^A(a_1, \ldots, a_n) = f^{\mathsf{A}'}(a_1, \ldots, a_n)$, else $f^A(a_1, \ldots, a_n)$ is undefined.
  - for any $h \in dom(\mu^P{}_0)_\Sigma(\mathsf{A}', \mathsf{B}')$ the arrow $\mu^P{}_\Sigma(\mathsf{h}')$ is $\mathsf{h}'_{|\mu^P{}_\Sigma(\mathsf{A}')}$. □

Now initial models are translated along $\mu^P{}_0$ to initial models; the proof follows a pattern common to most algebraic frames. First it is shown that the translation I of an initial object is *weakly* initial (i.e. that there exists at least one arrow from I into any object), so that the *no-confusion* condition holds; the weak initiality comes from $dom(\mu^P{}_0)_\Sigma$ being a full subcategory of the total models and $\mu^P{}_0$ being surjective on the objects. Then I is shown to be term-generated, so that the *no-junk* condition holds, too, because the total initial object is term-generated and term-generatedness is preserved by $\mu^P{}_0$.

Abstracting from the two main points of the above proof technique, we can define the *categorical* simulations, which preserve "term-generatedness" and whose domains are full subcategories, and show that categorical simulations preserve initiality.

Def. 2.7 Let $\mathbf{C}$ be a category and c be an object of $\mathbf{C}$; then c is called *inductive* iff $\mathbf{C}(c, c')$ has at most one element for every $c' \in \mathbf{C}$. For every subcategory $\mathbf{C}'$ of $\mathbf{C}$, c is called *weakly initial* in $\mathbf{C}'$ iff $\mathbf{C}'(c, c')$ has at least one element for every $c' \in \mathbf{C}'$.

Let $\mathcal{I}$ and $\mathcal{I}'$ be institutions and μ be a simulation from $\mathcal{I}$ into $\mathcal{I}'$. Then μ is called *categorical* iff every $dom(\mu)_\Sigma$ is a full sub-category of $Mod'(\Sigma)$ and μ_Σ preserves the inductive objects of $Mod'(\mu(\Sigma))$ belonging to $dom(\mu)_\Sigma$. □

Note that the property of being categorical only involves the model components of simulations (and, implicitly, the translation of signatures); thus if two simulations coincide on signatures and models and the first is categorical, then also the second one is so, independently of the formulas that are chosen as sentences of the institutions and their translation.

In most algebraic frames the interesting classes of models are closed w.r.t. subalgebras and this guarantees that their initial models, if any, are term-generated; this can be generalized to every categorical frame, noting that the notion of subalgebra generalizes to the categorical concept of *regular subobject*. Let us first recall the definition of regular sub-object.

Def. 2.8 Let $\mathbf{C}$ be a category and $f, g \in \mathbf{C}(A, B)$ be a pair of parallel arrows. Then an arrow $e \in \mathbf{C}(E, A)$ is an *equalizer* of f and g iff it satisfies the following conditions

- $f \cdot e = g \cdot e$ (*e equalizes f* and g);
- for any $k \in \mathbf{C}(K, A)$ s.t. $f \cdot k = g \cdot k$ there exists a unique $\pi \in \mathbf{C}(K, E)$ s.t. $e \cdot \pi = k$ (*k factorizes trough e*).

If $e \in \mathbf{C}(E, A)$ is an equalizer of some f and g, E is a *regular subobject* of A. □

Lemma 2.9 Let $\mathbf{C}$ be a category having equalizers and $\mathbf{C}'$ be a subcategory of $\mathbf{C}$ closed under equalizers and regular subobjects. Then I is initial in $\mathbf{C}'$ only if I is inductive in $\mathbf{C}$. □

It is worth to note that both the institutions of partial as well as total many-sorted algebras, with or without predicates, have equalizers, and that the model classes of positive Horn-clauses are closed w.r.t. regular subobjects (in general they are not closed w.r.t. generic subobjects); thus the following proposition applies in most cases.

Prop. 2.10 Let $\mathcal{I} = (\mathbf{Sign}, Sen, Mod, \models)$ and $\mathcal{I}' = (\mathbf{Sign}', Sen', Mod', \models')$ be institutions s.t. for any $\Sigma' \in |\mathbf{Sign}'|$ the category $Mod'(\Sigma')$ has equalizers and μ be a categorical simulation from $\mathcal{I}$ into $\mathcal{I}'$. If I' is initial in a full subcategory $\mathbf{C}'$ of $dom(\mu)_\Sigma$ closed w.r.t. regular subobjects (performed in $Mod'(\mu(\Sigma))$), then $\mu_\Sigma(I')$ is initial in $\mu_\Sigma(\mathbf{C}')$. □

Cor. 2.11 Let $\mathbf{C}'$ be a full subcategory of $dom({\mu^P}_0)_\Sigma$ closed w.r.t. subalgebras and I' be the initial object in $\mathbf{C}'$. Then ${\mu^P}_0(\mathsf{I}')$ is initial in ${\mu^P}_0(\mathbf{C}')$. □

Logical Level. Let us consider now the logical aspect of partial and total frames and investigate the equivalences of their expressive power. In the total [partial] frame we consider, as usual, the institution $\mathcal{TL}$ [$\mathcal{EPAR}$] of positive Horn-clauses [built on existential equality] and its subinstitution $\mathcal{GTL}$ [$\mathcal{GEPAR}$] where the sentences are without variables.

Let us consider first the trivial simulation $\mu_0^\perp$. Let A' be in $dom(\mu_0^\perp)_\Sigma$ and consider a ground existential equality $t = t'$; then $A = {\mu_0^\perp}_\Sigma(\mathsf{A}')$ satisfies $t = t'$ iff t^A and t'^A denote the same element of $s^A = s^{\mathsf{A}'} - \{\perp_s\}$, i.e. iff $t^{\mathsf{A}'} = t'^{\mathsf{A}'} \neq \perp_s$; thus to generalize $\mu_0^\perp$ to a simulation from $\mathcal{GEPAR}$, a stronger (and unusual) logic than the positive Horn-clauses is needed in the total frame. Therefore the trivial totalization fails in both the categorical and the logical aspects, in the sense that, although it is true that any partial algebra is equivalent from a set theoretic point of view to its trivial totalization, the equivalence becomes false if algebra morphisms are considered; moreover it relates Horn-Clauses to a more powerful first-order fragment.

Let us consider now the simulation ${\mu^P}_0$. Every ground Horn-clause is naturally translated into the total frame, just by replacing every existential equality symbol with the corresponding predicate eq. However if variables appear in the formula, this translation from the partial to the total frame does not preserve the validity of sentences. Indeed, consider for example $D_s(x)$; then obviously any partial algebra satisfies it (undefined elements do not exist), while some total algebras in the domain of ${\mu^P}_0$ do not, because valuations of variables in the total frame range also over the elements which do not satisfy the definedness predicates (and hence are dropped by the simulation). More generally the valuations for the total frame which range over undefined elements must not be taken in account, in order to establish the validity of translations of partial sentences. To overcome this problem it is sufficient to add to the premises of every sentence the definedness assertions for each of its variables, so that every valuation s.t. $V(x) = a$ and $\neg D(a)$ satisfies the sentence, because one of the premises is false; thus the validity only depends on "defined" valuations also in the total case. Note that in this way equations with variables in the partial frame are translated into conditional axioms of the total formalism. This, together with the fact that the simulation ${\mu^P}_0$ (properly generalized to deal with sentences) is categorical, illustrates the deep reason for the model classes of partial equational specifications being quasi-varieties (see [24]), like the model classes of total conditional specifications, and not varieties, as the model classes of total equational specifications are (see [14]).

Def. 2.12 The categorical simulation $\mu^P\colon \mathcal{EPAR} \to \mathcal{TL}$ coincides with ${\mu^P}_0$ on signatures and models, and on sentences is defined by

$$ {\mu^P}_\Sigma(\xi) = D_{s_1}(x_1) \wedge \ldots \wedge D_{s_k}(x_k) \wedge eq_{s'_1}(t_1, t'_1) \wedge \ldots \wedge eq_{s'_n}(t_n, t'_n) \supset eq_s(t, t') $$

where $\xi = (t_1 = t'_1 \wedge \ldots \wedge t_n = t'_n \supset t = t')$ and $x_1, \ldots, x_k$ are the variables of ξ. □

Since the domain of μ^P is the model class of the following axioms $th(\mu^P)$:

$$ D_s(f(x_1, \ldots, x_n)) \supset D_{s_i}(x_i) \text{ for } i = 1 \ldots n \text{ (\textit{strictness}) and} $$

$$ D_s(x) \wedge D_s(y) \wedge x = y \Leftrightarrow eq_s(x, y), \text{ i.e.} $$

$D_s(x) \wedge D_s(y) \wedge x = y \supset eq_s(x, y)$, $eq_s(x, y) \supset D_s(x)$, $eq_s(x, y) \supset D_s(y)$ and $eq_s(x, y) \supset x = y$, every model class of a partial *presentation* (Σ, Ax) is simulated by the model class of the total presentation $(\mu^P(\Sigma), {\mu^P}_\Sigma(Ax) \cup th(\mu^P))$. We call *logical* this kind of simulation, i.e. simulations translating presentations into presentations.

Def. 2.13 Let $\mathcal{I} = (\mathbf{Sign}, Sen, Mod, \models)$ and $\mathcal{I}' = (\mathbf{Sign}', Sen', Mod', \models')$ be institutions and μ be a simulation from $\mathcal{I}$ into $\mathcal{I}'$. Then μ is called *logical* iff it is categorical and $dom(\mu)_\Sigma$ is the model class of a set $th(\mu)_\Sigma \subseteq Sen'(\mu(\Sigma))$ of sentences for every $\Sigma \in |\mathbf{Sign}|$. □

Although in general the family $\{th(\mu)_\Sigma\}_{\Sigma \in |\mathbf{Sign}|}$ is not functorial, i.e. it is not possible to define a sub-functor F of Sen s.t. $F(\Sigma) = th(\mu)_\Sigma$, if we consider the family of the closures under logical consequences $\{th(\mu)^\bullet_\Sigma\}_{\Sigma \in |\mathbf{Sign}|}$, where $th(\mu)^\bullet_\Sigma = \{\alpha \mid A' \models' \alpha \quad \forall A' \in dom(\mu)_\Sigma\}$, then the partial-naturality condition on $\{dom(\mu)_\Sigma\}_{\Sigma \in |\mathbf{Sign}|}$ guarantees the functoriality of $\{th(\mu)^{\bullet}_{\Sigma}\}_{\Sigma \in |\mathbf{Sign}|}$ and hence every logical simulation is a map of institutions, too (see [13]). On the converse any surjective map of institutions is a logical simulation. In the next section logical simulations are also used to tailor inference systems to the domain of simulations.

Since the translations via μ^P of ground partial Horn Clauses are ground Horn Clauses, μ^P can be specialized to a simulation between the institutions $\mathcal{GEPAR}$ and $\mathcal{GTL}$. It is still categorical, but is not logical anymore; indeed axioms with variables are needed to define the domain, as the following example shows.

Example 2.14 For any signature $\Sigma = (S, F) \in |\mathbf{Sign}_{\mathcal{EPAR}}|$ with at least one function symbol there does not exist a set $th' \subseteq Sen'(\mu^P(\Sigma))$ of ground sentences s.t. $dom(\mu^P)_\Sigma$ is the class of models of th'. Indeed there exists an algebra A' which belongs to the model class of any set of ground Horn-clauses but does not to $dom(\mu^P)_\Sigma$. Let A' be defined by $s^{\mathsf{A}'} = \{1_s, 2_s\}$ for all $s \in S$, $f^{\mathsf{A}'}(x_1, \ldots, x_n) = 1_s$ for all $f \in F_{s_1 \ldots s_n, s}$, $eq_s^{\mathsf{A}'} = \{(1_s, 1_s)\}$ and $D_s{}^{\mathsf{A}'} = \{1_s\}$. Then for any ground term $t \in T_{\Sigma s}$, its evaluation in A' is 1_s and hence A' satisfies any ground formula. But A' does not belong to $dom(\mu^P)_\Sigma$, because functions are not strict; indeed $D_s{}^{\mathsf{A}'}(f^{\mathsf{A}'}(2_{s1}, \ldots, 2_{sn}))$ but $\neg D_{s_i}{}^{\mathsf{A}'}(2_{s_i})$. □

3 Inference System Translation

In this section we show how to translate inference systems via simulation in such a way that soundness and completeness of inference systems are preserved, and then apply this technique to a few specific examples and show that the results of [18] and of [4, 5] are instances of more general properties. Some more applications may be found in [11], where translations of partial, of Horn-clauses and of Order-sorted logics in terms of equational type logic are presented in order to use the ET-inference system and the connected rewrite tools; although these translations are not formalized as simulations, they can be so and the results obtained by their applications are an instance of the ones presented in the next subsection.

3.1 General Results

According to the intuition that a simulation codes a new institution in terms of an old one, inference systems are translated via simulation; so that, starting from an inference system for $\mathcal{I}'$ and using a simulation $\mu: \mathcal{I} \rightarrow \mathcal{I}'$, a new system for $\mathcal{I}$ is built, which consists of: the preprocessing μ of the sentences of $\mathcal{I}$, coding them as sentences of $\mathcal{I}'$, followed by the application of the given system for $\mathcal{I}'$, and possibly by the postprocessing μ^{-1} to decode the results.

Def. 3.1 Let $\mathcal{I} = (\mathbf{Sign}, Sen, Mod, \models)$ be an institution and $\vdash$ be an inference system for $Sen(\Sigma)$, i.e. any relation $\vdash \subseteq \wp(Sen(\Sigma)) \times Sen(\Sigma)$. Then $\vdash$ is *sound* for $C \subseteq |Mod(\Sigma)|$ iff for any $\phi \in Sen(\Sigma)$ and any $\Gamma \subseteq Sen(\Sigma)$, $\Gamma \vdash \phi$ implies that for all $A \in C$ if $A \models_\Sigma \gamma$ for all $\gamma \in \Gamma$, then $A \models_\Sigma \phi$. If C is $|Mod(\Sigma)|$, then $\vdash$ is shortly said sound.

For any $\Psi \subseteq Sen(\Sigma)$ and any $C \subseteq |Mod(\Sigma)|$, the system $\vdash$ is *complete* w.r.t. Ψ and C iff for any $\psi \in \Psi$ and any $\Gamma \subseteq Sen(\Sigma)$

$$A \models_\Sigma \gamma \text{ for all } \gamma \in \Gamma \text{ implies } A \models_\Sigma \psi \quad \text{for any } A \in C$$

implies $\Gamma \vdash \psi$. If C is $|Mod(\Sigma)|$, then $\vdash$ is shortly said complete w.r.t. Ψ.

For any simulation $\mu: \mathcal{I} \to \mathcal{I}'$ and any inference system $\vdash'$ for $Sen'(\mu(\Sigma))$, the inference system $\vdash'^\mu$ for $Sen(\Sigma)$ is defined by: $\Gamma \vdash'^\mu \phi$ iff $\mu(\Gamma) \vdash' \mu(\phi)$. □

The definition of completeness as it stands is a generalization of the notion of completeness in algebraic frames; indeed, for examples, in the frame of (both partial and total) conditional specifications the *equational completeness* of a system $\vdash$ means that if an equation $t = t'$ holds in the model class of a set of conditional axioms Γ, then $\Gamma \vdash t = t'$. Thus the premises Γ are any set of sentences, while the consequence has to be an equation, i.e. a sentence in the selected subclass.

Note that if reflexivity, monotonicity and transitivity are required by the definition of inference system, as for the *entailment systems* of [13], then simulations preserve these properties, so that the translation of an entailment system is an entailment system, too.

Prop. 3.2 Let $\mathcal{I} = (\mathbf{Sign}, Sen, Mod, \models)$ and $\mathcal{I}' = (\mathbf{Sign}', Sen', Mod', \models')$ be institutions, $\mu: \mathcal{I} \to \mathcal{I}'$ be a simulation and $\vdash'$ be an inference system for $Sen'(\mu(\Sigma))$.

1. if $\vdash'$ is *reflexive*, i.e. $\{\phi'\} \vdash' \phi'$, then $\vdash'^\mu$ is reflexive, too;
2. if $\vdash'$ is *monotonic*, i.e. $\Gamma'_1 \vdash' \phi'$ and $\Gamma'_1 \subseteq \Gamma'_2$ imply $\Gamma'_2 \vdash' \phi'$, then $\vdash'^\mu$ is monotonic;
3. if $\vdash'$ is *transitive*, i.e. $\Gamma' \vdash' \phi'_i$ for all $i \in I$ and $\{\phi'_i \mid i \in I\} \vdash' \phi'$ imply $\Gamma \vdash' \phi'$, then $\vdash'^\mu$ is transitive, too.
4. if $\vdash$ is *compact*, i.e. $\Gamma' \vdash' \phi'$ implies that there exists a finite $\Gamma'_1 \subseteq \Gamma'$ s.t. $\Gamma'_1 \vdash' \phi$, then $\vdash'^\mu$ is compact, too. □

Cor. 3.3 Let $\mathcal{I}' = (\mathbf{Sign}', Sen', Mod', \models')$ be an institution and $\vdash' = \{\vdash'_{\Sigma'} \mid \Sigma' \in |\mathbf{Sign}'|\}$ be an *entailment system* for $\mathcal{I}'$ (see definition 1 of [13]), i.e. a family of reflexive, monotonic and transitive relations $\vdash'_{\Sigma'} \subseteq \wp(Sen'(\Sigma')) \times Sen'(\Sigma')$ satisfying the following condition

* if $\Gamma' \vdash'_{\Sigma'_1} \phi'$, then for every $\sigma' \in \mathbf{Sign}'(\Sigma'_1, \Sigma'_2)$, $Sen'(\sigma')(\Gamma') \vdash_{\Sigma'_2} Sen'(\sigma')(\phi')$.

Then for any institution $\mathcal{I} = (\mathbf{Sign}, Sen, Mod, \models)$ and any simulation $\mu: \mathcal{I} \to \mathcal{I}'$ the family $\vdash'^\mu = \{\vdash'^\mu_\Sigma = \vdash'^\mu_{\mu(\Sigma)} \mid \Sigma \in |\mathbf{Sign}|\}$ is an entailment system for $\mathcal{I}$. □

The properties of simulations guarantee that if a system $\vdash'$ in the old institution $\mathcal{I}'$ is sound and complete w.r.t. the domain of the simulation μ, then the obtained system $\vdash'^\mu$ is sound and complete for $\mathcal{I}$, too, as the following theorem shows.

Theorem 3.4 Let $\mathcal{I}$ and $\mathcal{I}'$ be institutions, $\mu: \mathcal{I} \to \mathcal{I}'$ be a simulation and $\vdash'$ be an inference system for $Sen'(\mu(\Sigma))$;

1. if $\vdash'$ is sound for $|dom(\mu)_\Sigma|$, then $\vdash'^{\mu}$ is sound, too;
2. if $\vdash'$ is complete for $C' \subseteq |dom(\mu)_\Sigma|$ and $\Psi' \subseteq Sen'(\mu(\Sigma))$, then $\vdash'^{\mu}$ is complete for $\mu_\Sigma(C')$ and $\mu^{-1}(\Psi')$. □

Note that if a system $\vdash'$ is sound for $Mod'(\mu(\Sigma))$, then it is sound for any of its subcategories and hence any general system for $\mathcal{I}'$ is sufficient, if only soundness matters; but if completeness is considered too, then the system $\vdash'$ is required to be complete for the domain of the simulation, which is a subclass of the whole model class, and hence in general needs not be even sound for the whole model class. Thus in general $\vdash'$ is not a general system for the new institution, but it is tailored to the simulation, contrary to the intuition that simulations translate general results from one formalism to another. However, if the domain of μ coincides with the model class of some set th' of sentences, and hence in particular if μ is logical, then starting from any sound and complete inference system w.r.t. the whole class of models, we can apply the above theorem to the system $\vdash'_{th'}$, defined by $\Phi \vdash'_{th'} \phi$ iff $\Phi \cup th \vdash' \phi$, thus recovering the desired level of generality.

Cor. 3.5 Let $\mathcal{I}$ and $\mathcal{I}'$ be institutions, $\mu: \mathcal{I} \to \mathcal{I}'$ be a simulation s.t. $dom(\mu)_\Sigma = \{A' \mid A' \models'_{\mu(\Sigma)} \alpha', \alpha' \in th'\}$ for some $th' \subseteq Sen'(\mu(\Sigma))$ and $\vdash'$ be an inference system for $Sen'(\mu(\Sigma))$, which is sound and complete w.r.t. $\Psi' \subseteq Sen'(\mu(\Sigma))$. Then $\vdash'^{\mu}_{th'}$ is sound and complete w.r.t. $\mu^{-1}(\Psi')$, where $\vdash_{th'}$ denotes the system defined by $\Phi' \vdash_{th'} \phi'$ iff $\Phi' \cup th' \vdash \phi'$. □

So far the main interest was on the translation of inference systems from the old into the new frame; however, note that soundness and completeness are preserved in the opposite direction, too.

3.2 Applications

Many-sorted and Untyped Logic. As it was first pointed out in [14], the Birkhoff calculus for equational logic trivially generalized to the many-sorted case is not sound, if empty carriers are allowed. To solve this problem there are two main approaches: changing the notion of validity so that the Birkhoff calculus is sound also for the many-sorted case (see e.g. section 1 of [18] and of [10]), and using more sophisticated inference systems (see e.g. [14]). Since there exists the simulation $\mu^M: \mathcal{MS} \to \mathcal{L}$ of many-sorted by classical logic, applying the above Corollary 3.5, any sound and complete inference system for the classical logic may be translated into a sound and equationally complete system for the many-sorted calculus. Consider for example the homogeneous Birkhoff system, which consists of axioms for the equality to be a congruence and for substitution, enriched by the modus ponens rule.

Prop. 3.6 The Birkhoff system is sound and complete w.r.t. the set GEq of the ground equations $\{t = t' \mid t, t' \in T_\Sigma\}$, i.e. for any set Γ of conditional sentences and any ground equation $t = t'$

$$\Gamma \vdash t = t' \iff (\mathtt{A} \models_{\mathcal{L}} t = t' \quad \forall \mathtt{A} \quad s.t.\ \mathtt{A} \models_{\mathcal{L}} \gamma \quad \forall \gamma \in \Gamma). \ \square$$

Cor. 3.7 The translation $\vdash'^{\mu^M}_{th(\mu^M)}$ of the Birkhoff system along μ^M is sound and equationally complete, i.e. for any many-sorted ground equation $t = t'$ and any set Ax of many-sorted conditional equations $Ax \vdash'^{\mu^M}_{th(\mu^M)} \emptyset.t = t'$ iff ($\mathsf{A} \models_{\mathcal{MS}} \alpha$ for all $\alpha \in Ax$ implies $\mathsf{A} \models_{\mathcal{MS}} \emptyset.t = t'$) for every many-sorted algebra A, where $\emptyset$ denotes the empty type assignment to the empty set of variables. $\square$

Thus, because of the above corollary we have an equational calculus for every set Ax of conditional formulas in the many sorted frame, that consists of

Preprocessing $x_1 : s_1 \wedge \ldots \wedge x_k : s_k \wedge t_1 = t'_1 \wedge \ldots \wedge t_n = t'_n \supset t = t'$ for all $\alpha = (V.t_1 = t'_1 \wedge \ldots \wedge t_n = t'_n \supset t = t') \in Ax$ on variables $\{x_1, \ldots, x_n\}$ of type $V(x_i) = s_i$.

Apply the Birkhoff system (with default *well formedness* axioms for every $op \in F_{s_1 \ldots s_k, s}$) $\quad x_1 : s_1 \wedge \ldots \wedge x_k : s_k \supset op(x_1, \ldots, x_k) : s$

Postprocessing Output $\emptyset.t = t'$ for all deduced $t = t'$.

Note that, since the image along μ^M of an open equality is a conditional axiom, to show that the above calculus is complete w.r.t. *non-ground* equalities, we should first prove that the Birkhoff system is complete w.r.t. sentences of the form $x_1 : s_1 \wedge \ldots \wedge x_n : s_n \supset t = t'$, while the equational completeness is not sufficient.

In sections 2 and 3 of [18] it is shown, implicitly using the simulation μ^M and proving a subset of the results of the first subsection for this particular case, that the [14] equational calculus is an optimized version of a classical first-order inference system, so that soundness and completeness may be derived from the results of homogeneous first-order logic. Although the direct proofs of soundness and completeness of the Meseguer-Goguen system are not difficult, we think that the existence of a simulation, μ^M, which relates this system to (one version of) the classical Birkhoff system, enlightens the value of the Meseguer-Goguen results, showing that their system is not just a technical trick to overcome the empty carrier problem, but is also an elegant application of general results known from the homogeneous case.

Partial Higher-order into First-order Types. Recently higher-order specifications have become a standard tool in algebraic specifications, with a particular interest in the specification of partial higher-order functions. Higher-order functional spaces can be handled using the usual first-order algebraic specifications (see e.g. [15]), by restricting the signatures (S, F) to the ones where S is a subset of a set of functional sorts (i.e. $S \subseteq B^{\rightarrow}$, where $B^{\rightarrow}$ is inductively defined by $B \subseteq B^{\rightarrow}$ and $s_1 \ldots s_{n+1} \in B^{\rightarrow}$ implies $s_1 \times \ldots \times s_n \rightarrow s_{n+1} \in B^{\rightarrow}$) s.t. for every $(s_1 \times \ldots \times s_n \rightarrow s_{n+1}) \in S$ an explicit application operator belongs to the signature; moreover the models are required to be extensional, i.e. two elements of a functional sort yielding the same result on every input have to be equal. From a logical point of view, in the total case (see [12, 19]) an equationally complete system for the higher-order models may be obtained by enriching any (first-order) equationally complete system by the rule

$$* \frac{f(x_1, \ldots, x_n) = g(x_1, \ldots, x_n)}{f = g} \quad f, g \text{ terms of sort } (s_1 \times \ldots \times s_n \rightarrow s_{n+1});\ x_i \text{ variable of sort } s_i \text{ not appearing in } f \text{ and } g.$$

Instead in the partial case the above rule $*$ is insufficient to achieve a complete system; for a detailed discussion of this point see [4, 5]. However there is a logical simulation based on a skolemization procedure of higher-order by strongly conditional partial algebras (i.e. partial algebras with Horn-clauses based on both existential and *strong* equalities, where a strong equality holds iff either the existential equality holds or both sides are undefined), so that Corollary 3.5 applies and hence an equationally complete system for the higher-order models may be simulated by any equationally complete system for strongly conditional partial models. The intuition of this construction is that for each couple f, g of distinct functional elements a *witness* of their difference, i.e. an input (tuple) a s.t. $f(a) \neq g(a)$, exists; thus it is sufficient to introduce function symbols to denote the witnesses.

Def. 3.8 Let $\mathcal{PAR}$ be the institution of partial algebras with strongly conditional formulas as sentences (for references see [1, 3]) and $\mathcal{PHO}$ be the institution of extensional partial algebras on higher-order signatures with strongly conditional formulas as sentences, too (for references see [5]). Let $\mu^E: \mathcal{PHO} \rightarrow \mathcal{PAR}$ be the simulation consisting of:

- $\mu^E: \mathbf{Sign}_{\mathcal{PHO}} \rightarrow \mathbf{Sign}_{\mathcal{PAR}}$ is defined by $\mu^E((S, F)) = (S', F')$, where $S = S'$ and $F' = F \cup_{s=(s_1 \times \ldots \times s_n \rightarrow s_{n+1}) \in S} \{x_{s,i}: s \times s \rightarrow s_i \mid i = 1, \ldots, n\}$ and $\mu^E((\sigma, \phi)) = (\sigma, \phi')$, where $\phi'(f) = \phi(f)$ for all $f \in F$ and $\phi(x_{s,i}) = x_{\sigma(s),i}$.
- $\mu^E: Sen_{\mathcal{PHO}} \rightarrow Sen_{\mathcal{PAR}} \cdot \mu^E$ is the embedding natural transformation.
- $\mu^E: Mod_{\mathcal{PAR}} \cdot \mu^E \rightarrow Mod_{\mathcal{PHO}}$ is defined by
  - $dom(\mu^E)_\Sigma$ is the full subcategory of $Mod_{\mathcal{PAR}}(\mu^E((\sigma, \phi)))$ whose objects are the partial algebras A which satisfy the set $th(\mu^E)$ of axioms $\forall f, g : s.f(x_{s,1}(f,g), \ldots, x_{s,n}(f,g)) = g(x_{s,1}(f,g), \ldots, x_{s,n}(f,g)) \supset f = g$ for all $(s_1 \times \ldots \times s_n \rightarrow s_{n+1}) \in S$
  - Let $\iota: \Sigma \rightarrow \mu^E(\Sigma)$ be the signature embedding; then $\mu^E(A') = Mod_{\mathcal{PHO}}(\iota)(A')$ and $\mu^E(h') = Mod_{\mathcal{PHO}}(\iota)(h')$. In the following we denote $Mod_{\mathcal{PHO}}(\iota)(A')$ by $A'_{|\Sigma}$ and $Mod_{\mathcal{PHO}}(\iota)(h')$ by $h'_{|\Sigma}$

Since μ^E is the identity and μ^E is a family of forgetful functors, it is quite easy to check that μ^E is a simulation; the only non-trivial step is to check that μ^E is surjective on the objects, i.e. that for any extensional algebra A there exists an expansion A' of A (i.e. an algebra A' s.t. $A'_{|\Sigma} = \mathsf{A}$) s.t. $A' \in |dom(\mu^E)_\Sigma|$. Hence we only have to define $x_{s,i}^{A'}$ on A in such a way that A' satisfies the axioms

$$\forall f, g : s.f(x_{s,1}(f,g), \ldots, x_{s,n}(f,g)) = g(x_{s,1}(f,g), \ldots, x_{s,n}(f,g)) \supset f = g$$

Since A is extensional, for all $s = s_1 \times \ldots \times s_n \rightarrow s_{n+1}$ and all $\phi, \psi \in s^{\mathsf{A}}$ either $\phi = \psi$ or there exist $a_i \in s_i^{\mathsf{A}}$ for $i = 1 \ldots n$ s.t. $\phi(a_1, \ldots, a_n) \neq \psi(a_1, \ldots, a_n)$; in the first case let $x_{s,i}^{A'}(\phi, \psi)$ be undefined for $i = 1 \ldots n$, in the second one let $x_{s,i}^{A'}(\phi, \psi)$ be such an a_i for $i = 1 \ldots n$. Then it is easy to check that $A' \in dom(\mu^E)_\Sigma$. $\square$

Prop. 3.9 Let CL be the inference system for the partial strongly conditional logic, defined in [3]. The translation $\vdash'^{\mu^E}_{th(\mu^E)}$ of the CL system along μ^E is sound and equationally complete, i.e. for any either strong or existential equality ϵ on variables X and any set Ax of conditional formulas $Ax \vdash'^{\mu^E}_{th(\mu^E)} D(X) \supset \epsilon$ iff for every higher-order partial algebra A

$$\mathsf{A} \models_{\mathcal{PHO}} \alpha \text{ for all } \alpha \in Ax \text{ implies } \mathsf{A} \models_{\mathcal{PHO}} D(X) \supset \epsilon. \ \square$$

4 Related Work

Similar concepts. Due to the relevance of the interaction of different formal systems, several attempts to "put together institutions" have been developed and are still under development; let us summarize some related works. In the sequel we use, as a convention, "new" to denote the elements of the source of any arrow between institutions (independently from the direction of its components) and "old" to denote the element of the target. The first notion of arrow between institutions is that of *institution morphism*, introduced in [8]. Institution morphisms capture the idea of enriching an institution by new features and are mainly used to define *duplex* institutions, where sentences from an institution and constraints from another one are both available. Technically, morphisms differ from simulations, because they translate signatures and models covariantly, and sentences in the opposite direction. A closer (to simulations) notion of institution arrow is the *coding*, presented in the draft [25] to investigate on the expressive power of LF (Edinburgh Logical Framework). Indeed the direction of the components of coding is the same as the one of simulation and the philosophy is that the old model class can be partitioned in subclasses, each one representing some new model, in the sense that a class satisfies (the translation of) the same formulas which are satisfied by the represented model. Thus, from a technical point of view, there are three main differences: the model component is total, it is non-necessarily surjective, and the satisfaction preserving condition is between a new model and the whole class of its old representations. The main issue of [24] is "putting together" representation of logics in the common frame of the LF-institution. A quite close notion is that of *map* of [8] by Meseguer (see [13]), which follows the same intuition as logical simulations: maps relate new specifications to old ones and models are consistently translated from the old into the new frame; however the two notions are not exactly the same. Indeed maps of institutions are not required to be surjective (but if they are, then are also logical simulations); on the converse logical simulations are map. Although the two notions are strictly related, they are used for different purposes; indeed in [13] the focus is on the logical side, so that tools are introduced to deal with entailment systems, proofs and proof calculi, and then applied to propose semantics for logic programming, while the semantic side is quite neglected. It is still under development (see [21]) a new tool, called *transformation*, to relate pre-institutions (which are institutions where the satisfaction condition and the categorical structure of models have been dropped) and hence, in particular, institutions. Transformations translate every new sentence into a set of old sentences and every new model into a set of old models, requiring that a sentence is satisfied by a model iff its translation is satisfied by the class of models which is the translation of the model. In [21] the different levels of pre-institutions and transformations between them are analysed and some results are presented that are strictly connected to classical logic, like compactness theorems.

From institution independence to simulation independence The modularity principle applied to algebraic specifications requires that large specifications may be built starting from smaller ones, using specification languages, like *ASL* and *Clear*. The first attempt to generalize specification languages abstracting away from the frame chosen to define basic specifications, is the concept of *institution independent* language, proposed and illustrated on a significant example by Sannella and Tarlecki in [22], where, to build specifications in a uniform way w.r.t. the adopted frame, operations are defined using only the elements common to every institution, like signatures, models and sentences, and mathematical constructions on them. In [2], adding to the work in [22], *simulation-independent* languages are introduced, which are institution independent languages s.t. any simulation behaves as a homomorphism w.r.t. them; thus for any simulation-independent language the input specifications can be defined in different frames and then translated into a common frame, where the specification building operation is performed, and the result is independent from the chosen "super"frame, in the sense that the translation via simulation of the result into any other frame is equal to the result of the operation in the other frame on the translation of the inputs.

Third dimension of implementation Implementation, or, better, refinement (see [23]), has two directions of composition: vertical (refinement of refinement is refinement, too) and horizontal (if a parameterized specification sp_1 is a refinement of sp_2 and an actual parameter p_1 is a refinement of p_2, then the application $sp_1(p_1)$ is a refinement of $sp_2(p_2)$). Using simulations, a third direction is added (see [2]); indeed specifications defined in the old institution are implementations of their translations in the new and the three compositions are compatible.

Acknowledgements. We would like to thank J. Goguen, J. Meseguer, D. Sannella, P. Scollo and A. Tarlecki for the fruitful discussions on the subject and the referees for pointing out some delicate points.

References

1. E. Astesiano and M. Cerioli. On the existence of initial models for partial (higher-order) conditional specifications. In *TAPSOFT'89*, number 351 in Lecture Notes in Computer Science, pages 74–88, Berlin, 1989. Springer Verlag.
2. E. Astesiano and M. Cerioli. Commuting between institutions via simulation. Technical report, University of Genova, 1990.
3. E. Astesiano and M. Cerioli. Free objects and equational deduction for partial conditional specifications. Technical report, University of Genova, 1990.
4. E. Astesiano and M. Cerioli. Partial higher-order specifications. In *Math. Found. of Comp. Sci. '91*, number 520 in Lecture Notes in Computer Science, pages 74–84, Berlin, 1991. Springer Verlag.
5. E. Astesiano and M. Cerioli. Partial higher-order specifications. *Fundamenta Informaticae*, 16(2), 1992. To appear.
6. M. Broy and M. Wirsing. Partial abstract types. *Acta Informatica*, (18):47–64, 1982.
7. P. Burmeister. Partial algebras - survey of an unifying approach towards a two-valued model theory for partial algebras. *Algebra Universalis*, 15, 1982.
8. R. Burstall and J. Goguen. Introducing institutions. In E. Clarke and D. Kozen, editors, *Logics of Programming Workshop*, number 164 in Lecture Notes in Computer Science, pages 221–255, Berlin, 1984. Springer Verlag.
9. R. Burstall and J. Goguen. Institutions: Abstract model theory for specification and programming. Technical report, Computer Science Lab., SRI International, 1990.
10. V. Manca and A. Salibra. Equational calculi for many-sorted algebras with empty carriers. In *Math. Found. of Comp. Sci. '90*, number 452 in Lecture Notes in Computer Science, 1990.
11. V. Manca, A. Salibra, and G. Scollo. On the expressiveness of equational type logic. In C. Rattray and R. Clark, editors, *The Unified Computation Laboratory*. Oxford University Press, 1992.
12. K. Meinke. Universal algebra in higher types. *Theoretical Computer Science*, 1992.
13. J. Meseguer. General logics. In *Logic Colloquium '87*. North Holland, 1989.
14. J. Meseguer and J. Goguen. *Initiality, Induction and Computability*, pages 459–540. Cambridge University Press, Cambridge, 1985.
15. B. Möller. Algebraic specifications with higher-order operations. In *IFIP TC 2 Working Conference on Program Specification and Transformation*. North Holland, 1987.
16. P. Mosses. Unified algebras and modules. Technical Report DAIMI PB-266, CS Dept., Aarhus University, 1988.
17. M. Navarro, P. Nivela, F. Orejas, and A. Sanchez. On translating partial to total specifications with applications to theorem proving for partial specifications. Technical Report LSI-89-21, Universitat Politecnica de Catalunya, 1990.
18. P. Padawitz and M. Wirsing. Completeness of many-sorted equational logic revisited. *Bulletin EATCS*, (24), 1984.
19. Z. Qian. Higher-order order-sorted algebras. In *2nd International Conference on Algebraic and Logic Programming*, Lecture Notes in Computer Science, Berlin, 1990.
20. H. Reichel. *Initial Computability, Algebraic Specifications, and Partial Algebras*. Oxford University Press, 1987.

21. A. Salibra and G. Scollo. A soft stairway to institutions. In this volume, August 1991.
22. D. Sannella and A. Tarlecki. Specifications in an arbitrary institution. *Information and Computation*, (76):165–210, 1988.
23. D. Sannella and M. Wirsing. Implementations of parameterized specifications. In *9th Colloquium on Automata, Languages and Programming*, number 140 in Lecture Notes in Computer Science, Berlin, 1982. Springer Verlag.
24. A. Tarlecki. Quasi-varieties in abstract algebraic institutions. *Journal of Computer and System Science*, (33):333–360, 1986.
25. A. Tarlecki. Institution representation. draft note, November 1987.