

Borrowing Interpolation

Răzvan Diaconescu

Institute of Mathematics “Simion Stoilow” of the Romanian Academy

Abstract

We present a generic method for establishing interpolation properties by ‘borrowing’ across logical systems. The framework used is that of the so-called ‘institution theory’ which is a categorical abstract model theory providing a formal definition for the informal concept of ‘logical system’ and a mathematical concept of ‘homomorphism’ between logical systems. We develop three different styles or patterns to apply the proposed borrowing interpolation method. These three ways are illustrated by the development of a series of concrete interpolation results for logical systems that are used in mathematical logic or in computing science, most of these interpolation properties apparently being new results. These logical systems include fragments of (classical many sorted) first order logic with equality, preordered algebra and its Horn fragment, partial algebra, higher order logic. Applications are also expected for many other logical systems, including membership algebra, various types of order sorted algebra, the logic of predefined types, etc., and various combinations of the logical systems discussed here.

1. Introduction

The concept of institution arose within computing science (algebraic specification) in response to the population explosion among logics in use there, with the ambition of doing as much as possible at a level of abstraction independent of commitment to any particular logic [27, 34, 61]. Besides its extensive use in specification theory (it has become the most fundamental mathematical structure in algebraic specification theory), there have been several substantial developments towards an ‘institution-independent’ (abstract) model theory, [15, 18–20, 38, 39, 56, 57, 65, 66] being just few examples from a growing list of papers on institution-independent model theory. A monography [23] dedicated to this topic has been recently published. Recently institutions have also been extended towards proof theory [22, 53] in the spirit of categorical logic [43].

One of the most important methods in institution-independent model theory is that of ‘borrowing’ logical structure or properties across logical systems formalized as institutions. This method owes to the existence of a concept of homomorphism between institutions, in the literature called ‘comorphism’ [37]. Institution comorphisms are mappings which preserve the mathematical structure of institutions, but from a logic and model theoretic perspective their significance is that of either embeddings or encodings between institutions. An important feature of comorphisms is that they relate between institutions both at the syntactic and at the semantic level, and both the syntactic and the semantic components of comorphisms are coherent to each other. One of the early contributions in the area of borrowing of logical structure along institution comorphisms is [10].

Email address: Razvan.Diaconescu@imar.ro (Răzvan Diaconescu)

Our work extends the borrowing idea to interpolation. Interpolation is one of the most studied properties in mathematical logic [11, 63], a recent monography dedicated to interpolation in modal and intuitionistic being [33]. One of the reasons for the great interest in interpolation is perhaps the fact that it is the source for many other logical results and properties, a rather notorious case being that of definability. Interpolation has numerous applications in computing science especially in formal specification theory [5, 8, 27, 29, 30, 69]. but also in data bases (ontologies) [42], automated reasoning [54, 55], type checking [41], model checking [47], and structured theorem proving [1, 46]. Consequently it has also received special attention within institution-independent model theory [20, 21, 39]. In spite of its naturalness and simplicity, just saying that any deduction $\rho_1 \vdash \rho_2$ can be split as $\rho_1 \vdash \rho \vdash \rho_2$ such that ρ contains only symbols ‘common’ to both ρ_1 and ρ_2 , interpolation is notoriously difficult to obtain.

Here we develop a generic theorem for borrowing interpolation along institution comorphisms $\mathcal{I} \rightarrow \mathcal{I}'$. In brief, this means that if \mathcal{I}' has a certain interpolation property and the institution comorphism ‘behaves well’ with respect to interpolation, then \mathcal{I} can be established to have a corresponding interpolation property. This generic result can be applied to various situations, requiring different styles of applying it. One situation is when \mathcal{I} and \mathcal{I}' have essentially the same expressive power, such as (first order) partial algebra and first order logic. Although partial algebra is more refined than first order logic, it can be encoded into first order logic in a rather strong way which makes possible a quite straightforward transfer of first order logic interpolation to partial algebra. The second situation we study is when \mathcal{I} has significantly less expressive power than \mathcal{I}' , which means that their interpolation properties are rather different and moreover are obtained in different ways. This second situation is well illustrated by the logic of universal sentences versus first order logic, which is an embedding rather than an encoding. In this case the application of the main borrowing theorem requires more subtlety. Finally, a third situation is when \mathcal{I} has more expressive power than \mathcal{I}' but an encoding of \mathcal{I} into \mathcal{I}' by means of a comorphism exists. This is well illustrated by higher order logic in the role of \mathcal{I} and first order logic in the role of \mathcal{I}' .

Our general borrowing interpolation result, together with its above mentioned associated styles or patterns for applying it, constitute a new method within the rather large spectrum of methods to obtain interpolation. Taking into account the great difficulty of the problem, the existing methods are never enough. The strength of our borrowing method has at least several aspects to be mentioned. One aspect is that it has enabled us to obtain new (at least up to our knowledge) interpolation results (e.g. Corollaries 4.2, 4.3, 6.1) in a rather smooth way. For many of the applications the alternatives seem to be much more difficult. For example, in partial or preordered algebra, based on some similarity with classical many-sorted first order logic, we may expect certain interpolation properties. We either have to replicate the whole complicated conceptual infrastructure of the classical framework to these new frameworks (which is very complex and even problematic task) or else to lift such infrastructure to a higher abstraction level as in [20, 39] (which is also a difficult task but in a different way). Our method saves such efforts. Moreover in other situations (such as higher order logic) perhaps none of the above mentioned alternatives would work. Our borrowing method applies not only for obtaining new results but also for a better understanding of already know results (e.g. Corollaries 4.1, 5.1, 5.2). All these are related to a deeper aspect, that of offering a rather clear, although unavoidably partial, insight into the interdependent nature of interpolation properties in various logical systems.

The paper is structured as follows:

1. We recall the institution theoretic concepts necessary for the work developed in this paper.
2. We develop the main result of this paper represented by the generic borrowing interpolation theorem.
3. We develop three styles or patterns of applications of the main theorem, which have been mentioned above. We devote one section to each of these.

2. Institutions

This is a rather preliminary section which is structured as follows:

1. Since institution theory makes a rather intensive use of categorical concepts, we briefly recall the category theory necessary for our work.
2. We recall the definition of institution and introduce in some detail the examples of institutions which we will use in the development of the applications.
3. We recall the definition of institution comorphism.
4. We recall the general construction of the ‘institution of the presentations’ over a given (base) institution, construction which is used for the encodings between the institutions.

2.1. Categories

We assume the reader is familiar with basic notions and standard notations from category theory; e.g., see [45] for an introduction to this subject. The categorical notions required for this paper are rather basic, the reader is expected to have knowledge only of the concepts of category, functor, natural transformation, and pushout.

With respect to notational conventions, $|\mathbb{C}|$ denotes the class of objects of a category \mathbb{C} , $\mathbb{C}(A, B)$ the set of arrows (morphisms) with domain A and codomain B , and composition is denoted by “;” and in diagrammatic order. The category of sets (as objects) and functions (as arrows) is denoted by Set , and CAT is the category of all categories.¹

2.2. Institutions

Institutions have been defined by Goguen and Burstall in [9], the seminal paper [34] being printed after a delay of many years. Below we recall the concept of institution which formalises the intuitive notion of logical system, including syntax, semantics, and the satisfaction between them.

Definition 2.1 (Institutions). *An institution $\mathcal{I} = (\text{Sig}^{\mathcal{I}}, \text{Sen}^{\mathcal{I}}, \text{Mod}^{\mathcal{I}}, \models^{\mathcal{I}})$ consists of*

1. *a category $\text{Sig}^{\mathcal{I}}$, whose objects are called signatures,*
2. *a functor $\text{Sen}^{\mathcal{I}} : \text{Sig}^{\mathcal{I}} \rightarrow \text{Set}$, giving for each signature a set whose elements are called sentences over that signature,*
3. *a functor $\text{Mod}^{\mathcal{I}} : (\text{Sig}^{\mathcal{I}})^{\text{op}} \rightarrow \text{CAT}$ giving for each signature Σ a category whose objects are called Σ -models, and whose arrows are called Σ -(model) morphisms, and*
4. *a relation $\models_{\Sigma}^{\mathcal{I}} \subseteq |\text{Mod}^{\mathcal{I}}(\Sigma)| \times \text{Sen}^{\mathcal{I}}(\Sigma)$ for each $\Sigma \in |\text{Sig}^{\mathcal{I}}|$, called Σ -satisfaction,*

such that for each morphism $\varphi : \Sigma \rightarrow \Sigma'$ in $\text{Sig}^{\mathcal{I}}$, the satisfaction condition

$$M' \models_{\Sigma'}^{\mathcal{I}} \text{Sen}^{\mathcal{I}}(\varphi)(\rho) \text{ if and only if } \text{Mod}^{\mathcal{I}}(\varphi)(M') \models_{\Sigma}^{\mathcal{I}} \rho$$

holds for each $M' \in |\text{Mod}^{\mathcal{I}}(\Sigma')|$ and $\rho \in \text{Sen}^{\mathcal{I}}(\Sigma)$. We denote the reduct functor $\text{Mod}^{\mathcal{I}}(\varphi)$ by $_ \downarrow_{\varphi}$ and the sentence translation $\text{Sen}^{\mathcal{I}}(\varphi)$ by $\varphi(_)$. When $M = M' \downarrow_{\varphi}$ we say that M is a φ -reduct of M' , and that M' is a φ -expansion of M . When there is no danger of ambiguity, we may skip the superscripts from the notations of the entities of the institution; for example $\text{Sig}^{\mathcal{I}}$ may be simply denoted Sig .

¹Strictly speaking, this is only a quasi-category living in a higher set-theoretic universe.

General assumption: We assume that all our abstract institutions are such that satisfaction is invariant under model isomorphism, i.e. if Σ -models M, M' are isomorphic, denoted $M \cong M'$, then $M \models_{\Sigma} \rho$ if and only if $M' \models_{\Sigma} \rho$ for all Σ -sentences ρ . It will be easy to see that this assumption holds in all the concrete examples of institutions which appear in our paper.

In the following we give several examples of logics captured as institutions, which will be used in our applications.

Example 2.1. (Classical logic).

Let **FOL** be the institution of *first order logic with equality* in its many sorted form.

Its *signatures* are triples (S, F, P) consisting of

- a set of sort symbols S ,
- a family $F = \{F_{w \rightarrow s} \mid w \in S^*, s \in S\}$ of sets of function symbols indexed by arities (for the arguments) and sorts (for the results), and
- a family $P = \{P_w \mid w \in S^*\}$ of sets of relation (predicate) symbols indexed by arities.

Signature morphisms map the three components in a compatible way. This means that a signature morphism $\varphi: (S, F, P) \rightarrow (S', F', P')$ consists of

- a function $\varphi^{\text{st}}: S \rightarrow S'$,
- a family of functions $\varphi^{\text{op}} = \{\varphi_{w \rightarrow s}^{\text{op}}: F_{w \rightarrow s} \rightarrow F'_{\varphi^{\text{st}}(w) \rightarrow \varphi^{\text{st}}(s)} \mid w \in S^*, s \in S\}$, and
- a family of functions $\varphi^{\text{rl}} = \{\varphi_{w \rightarrow s}^{\text{rl}}: P_w \rightarrow P'_{\varphi^{\text{st}}(w)} \mid w \in S^*, s \in S\}$.

Models M for a signature (S, F, P) are first order structures interpreting each sort symbol s as a set M_s , each function symbol σ as a function M_{σ} from the product of the interpretations of the argument sorts to the interpretation of the result sort, and each relation symbol π as a subset M_{π} of the product of the interpretations of the argument sorts. In order to avoid the existence of empty interpretations of the sorts, which may complicate unnecessarily our presentation, we assume that each signature has at least one *constant* (i.e. function symbol with empty arity) for each sort. A model homomorphism $h: M \rightarrow M'$ is an indexed family of functions $\{h_s: M_s \rightarrow M'_s\}_{s \in S}$ such that

- h is an F -algebra homomorphism $M \rightarrow M'$, i.e., $h_s(M_{\sigma}(m)) = M'_{\sigma}(h_w(m))$ for each $\sigma \in F_{w \rightarrow s}$ and each $m \in M_w$, and
- $h_w(m) \in M'_{\pi}$ if $m \in M_{\pi}$ (i.e. $h_w(M_{\pi}) \subseteq M'_{\pi}$) for each relation $\pi \in P_w$ and each $m \in M_w$.

where $h_w: M_w \rightarrow M'_w$ is the canonical component-wise extension of h , i.e. $h_w(m_1, \dots, m_n) = (h_{s_1}(m_1), \dots, h_{s_n}(m_n))$ for $w = s_1 \dots s_n$ and $m_i \in M_{s_i}$. A model homomorphism is *closed* when $M_{\pi} = h_w^{-1}(M'_{\pi})$ for each relation symbol $\pi \in P_w$.

For each signature morphism φ , the *reduct* $M' \downarrow_{\varphi}$ of a model M' is defined by $(M' \downarrow_{\varphi})_x = M'_{\varphi(x)}$ for each sort, function, or relation symbol x from the domain signature of φ .

Sentences are the usual first order sentences built from equational and relational atoms by iterative application of Boolean connectives and quantifiers. Sentence translations along signature morphisms just rename the sorts, function, and relation symbols according to the respective signature morphisms. They can be formally defined by induction on the structure of the sentences. While the induction step is straightforward for the case of the Boolean connectives it needs a bit of attention for the case of the quantifiers. For any signature morphism $\varphi: (S, F, P) \rightarrow (S', F', P')$,

$$\text{Sen}^{\text{FOL}}(\varphi)((\forall X)\rho) = (\forall X^{\varphi})\text{Sen}^{\text{FOL}}(\varphi')(\rho)$$

for each finite set X of *variables for* (S, F, P) . The variables need to be disjoint from the constants of the signature, also we have to ensure that Sen^{FOL} thus defined is functorial indeed and that there is no overloading of variables (which in certain situations would cause a failure of the Satisfaction Condition). These may be formally achieved by considering that a variable for (S, F, P) is a triple of the form $(x, s, (S, F, P))$ where x is the *name of the variable* and $s \in S$ is the *sort of the variable* and that two different variables in X have different names. Then we let $(S, F + X, P)$ be the extension of (S, F, P) such that $(F + X)_{w \rightarrow s} = F_{w \rightarrow s}$ when w is non-empty and $(F + X)_{\rightarrow s} = F_{\rightarrow s} \cup \{(x, s, (S, F, P)) \mid (x, s, (S, F, P)) \in X\}$ and we let $\varphi' : (S, F + X, P) \rightarrow (S', F' + X^\varphi, P')$ be the canonical extension of φ that maps each variable $(x, s, (S, F, P))$ to $(x, \varphi(s), (S', F', P'))$.

The satisfaction of sentences by models is the usual Tarskian satisfaction defined inductively on the structure of the sentences.

Example 2.2. (Universal sentences).

A *universal sentence* for a **FOL** signature (S, F, P) is a sentence of the form $(\forall X)\rho$, where X is a set of (first order) variables and ρ is a quantifier-free $(S, F + X, P)$ -sentence, i.e. a sentence formed only from atoms by iterative applications of Boolean connectives only. Thus, the institution **UNIV** of the universal sentences has the same signatures and models as **FOL** but only the universal sentences as sentences. The institution **UNIV** can be obviously regarded as a ‘sub-institution’ of **FOL**, as we will see later this situation is an example of an institution comorphism.

Example 2.3. (Horn clauses).

A *Horn clause* is an universal sentence of the form $(\forall X)H \Rightarrow C$ where H is a finite conjunctions of atoms and C is a single atom. Let **HCL** denotes the sub-institution of **UNIV** determined by the Horn clauses.

Example 2.4. (Partial algebra).

Here we consider the institution **PA** of partial algebra as employed by the specification language CASL [4].

A *partial algebraic signature* is a tuple (S, TF, PF) , where TF is a family of sets of *total* function symbols and PF is a family of sets of *partial* function symbols such that $TF_{w \rightarrow s} \cap PF_{w \rightarrow s} = \emptyset$ for each arity w and each sort s . In order to avoid empty carriers, like in the case of **FOL**, we assume there exists at least one *total* constant for each sort. Signature morphisms map the three components in a compatible way.

A partial algebra is just like an ordinary algebra (i.e. a **FOL** model without relations) but interpreting the function symbols of PF as partial rather than total functions. A *partial algebra homomorphism* $h : A \rightarrow B$ is a family of (total) functions $\{h_s : A_s \rightarrow B_s\}_{s \in S}$ indexed by the set of sorts S of the signature such that $h_s(A_\sigma(a)) = B_\sigma(h_w(a))$ for each function symbol $\sigma \in TF_{w \rightarrow s} \cup PF_{w \rightarrow s}$ and each string of arguments $a \in A_w$ for which $A_\sigma(a)$ is defined.

The sentences have three kinds of atoms: *definedness* $\text{def}(t)$, *strong equality* $t \stackrel{s}{=} t'$, and *existence equality* $t \stackrel{e}{=} t'$. The definedness $\text{def}(t)$ of a term t holds in a partial algebra A when the interpretation A_t of t is defined. The strong equality $t \stackrel{s}{=} t'$ holds when both terms are undefined or both of them are defined and are equal. The existence equality $t \stackrel{e}{=} t'$ holds when both terms are defined and are equal.² The sentences are formed from these atoms by Boolean connectives and quantifications over total variables (i.e variables that are always defined).

Example 2.5. (Preordered algebra).

Preordered algebras are used for formal specification and verifications of algorithms [25], for automatic

²Notice that $\text{def}(t)$ is equivalent to $t \stackrel{e}{=} t$ and that $t \stackrel{s}{=} t'$ is equivalent to $(t \stackrel{e}{=} t') \vee (\neg \text{def}(t) \wedge \neg \text{def}(t'))$.

generation of case analysis [25], and in general about reasoning about transitions between states of systems. They constitute an unlabeled form of rewriting logic of [49]. Let **POA** denote the institution of preordered algebras.

The signatures are just ordinary *algebraic signatures* (S, F) , i.e. **FOL** signatures without the relation component. The **POA** models are *preordered algebras* which are interpretations of the signatures into the category of preorders $\mathbb{P}re$ rather than the category of sets $\mathbb{S}et$. This means that each sort gets interpreted as a preorder, and each function symbol as a monotonic function. A *preordered algebra homomorphism* is just a family of monotonic functions which is an algebra homomorphism.

The sentences have two kinds of atoms: (ordinary) equations $t = t'$ and preorder atoms $t \leq t'$. A preorder atom $t \leq t'$ is satisfied by a preordered model M when the interpretations of the terms are in the preorder relation of the carrier, i.e. $M_t \leq M_{t'}$. The sentences are formed from these atoms by Boolean connectives and quantifications over variables.

Example 2.6. (Horn preordered algebra).

This institution, denoted **HPOA**, is the Horn fragment of **POA**, i.e. the sub-institution of **POA** determined by the **POA** sentences of the form $(\forall X)H \Rightarrow C$ where H is a finite conjunction of atoms and C is a single atom. The CafeOBJ [25, 26] realization of preordered algebra is based upon this institution.

Example 2.7. (Higher order logic with Henkin semantics).

Higher order logic with Henkin semantics has been introduced and studied in [12, 40], a recent book on the topic being [3]. Here, in order to simplify the presentation and the illustration of our interpolation borrowing method, we consider an simplified variant close to the ‘higher order algebra’ of [51] which does not consider λ -abstraction and choice functions. Let us first define an institution of ‘plain’ higher order logic, denoted **HOL**.

For any set S of *sorts*, let \vec{S} be the set of S -*types* defined as the least set such that $S \subseteq \vec{S}$ and $s_1 \rightarrow s_2 \in \vec{S}$ when $s_1, s_2 \in \vec{S}$. A **HOL** *signature* (S, F) consists of a set S of sorts and a family of sets of constants $F = \{F_s\}_{s \in \vec{S}}$. A *morphism of HOL signatures* $\varphi: (S, F) \rightarrow (S', F')$ consists of a function $\varphi^{\text{st}}: S \rightarrow S'$ and a family of functions between operation symbols $\{\varphi_s^{\text{op}}: F_s \rightarrow F'_{\varphi^{\text{type}}(s)}\}_{s \in \vec{S}}$ where $\varphi^{\text{type}}: \vec{S} \rightarrow \vec{S}'$ is the extension of φ^{st} such that $\varphi^{\text{type}}(s_1 \rightarrow s_2) = \varphi^{\text{type}}(s_1) \rightarrow \varphi^{\text{type}}(s_2)$.

Given a signature (S, F) , an (S, F) -*model* interprets each sort $s \in S$ as a set M_s and each operation symbol $\sigma \in F_s$ as an element $M_\sigma \in M_s$, where for each types $s_1, s_2, M_{s_1 \rightarrow s_2} = [M_{s_1} \rightarrow M_{s_2}] = \{f \text{ function } | f: M_{s_1} \rightarrow M_{s_2}\}$. An (S, F) -*model homomorphism* $h: M \rightarrow N$ interprets each S -type s as a function $h_s: M_s \rightarrow N_s$ such that $h(M_\sigma) = N_\sigma$ for each $\sigma \in F$ and such that the diagram below

$$\begin{array}{ccc} M_s & \xrightarrow{f} & M_{s'} \\ h_s \downarrow & & \downarrow h_{s'} \\ N_s & \xrightarrow{h_{s \rightarrow s'}(f)} & N_{s'} \end{array}$$

commutes for all types s and s' and each $f \in M_{s \rightarrow s'}$.

For any **HOL** signature (S, F) , each operation symbol σ of type s is a term of type s , and (tt') is a term of type s_2 whenever t is a term of type $s_1 \rightarrow s_2$ and t' is a term of type s_1 . A **HOL** (S, F) -*equation* consists of a pair $t_1 = t_2$ of terms of the same type. A **HOL** (S, F) -*sentence* is obtained from equations by iteration of the usual Boolean connectives and of *higher order (universal or existential) quantification* which is defined

similarly to the quantification in **FOL**. Note however that because of the ‘higher order’ types, the constants in **HOL** denote higher order rather than first order entities.

The interpretation of operation symbols by models can be extended to terms by defining $M_{(tt')} = M_t(M_{t'})$ for each term t of type $s_1 \rightarrow s_2$ and each term t' of type s_1 . A model M satisfies the equation $t = t'$ when $M_t = M_{t'}$. This satisfaction relation can be extended in an obvious manner, similar to **FOL** or **PA**, from equations to any sentences.

The institution of *higher order logic with Henkin semantics*, denoted **HNK**, extends the classical models of a **HOL** signature by relaxing the condition $M_{s \rightarrow s'} = [M_s \rightarrow M_{s'}]$ to $M_{s \rightarrow s'} \subseteq [M_s \rightarrow M_{s'}]$.

Computing science abounds of examples of institutions, some being rather conventional, but most of them being rather unconventional from the viewpoint of the logic tradition. Some important institutions in use in computing science (other than those mentioned above) include membership [50], polymorphic [62], various modal logics such as temporal [32], process [32], behavioural [6], coalgebraic [13], object-oriented [36], multi-algebraic (non-determinism) [44], and pre-defined types [17] logics.

The following definitions introduce some very basic model theoretic concepts at the level of abstract institutions.

Definition 2.2. *For any signature Σ in an institution I :*

- For each set E of Σ -sentences, let $E^* = \{M \in \text{Mod}(\Sigma) \mid M \models_{\Sigma} e \text{ for each } e \in E\}$, and
- For each class \mathbb{M} of Σ -models, let $\mathbb{M}^* = \{e \in \text{Sen}(\Sigma) \mid M \models_{\Sigma} e \text{ for each } M \in \mathbb{M}\}$.

If E and E' are sets of sentences of the same signature, then $E' \subseteq E^*$ is denoted by $E \models E'$. Two sentences, ρ_1 and ρ_2 of the same signature are semantically equivalent, denoted \models when $\rho_1 \models \rho_2$ and $\rho_2 \models \rho_1$.

Definition 2.3 (Semantic Boolean connectives). *An institution has (semantic) conjunctions when for each signature Σ and any Σ -sentences e_1 and e_2 there exists a Σ -sentence e such that $e^* = e_1^* \cap e_2^*$. Usually e is denoted by $e_1 \wedge e_2$.*

It has (semantic) implications when for each e_1 and e_2 as above there exists e such that $e^ = (\text{Mod}(\Sigma) - e_1^*) \cup e_2^*$. Usually e is denoted $e_1 \Rightarrow e_2$.*

Definition 2.4 (Compactness). *An institution is compact when for each signature Σ , any set of Σ -sentences E and each Σ -sentence e , if $E \models_{\Sigma} e$ then there exists a finite set E_0 of Σ -sentences such that $E_0 \models_{\Sigma} e$.*

An institution is quasi-compact when it is compact or when it has infinite conjunctions.

Definition 2.5 (Conservative signature morphisms). *A signature morphism $\varphi: \Sigma \rightarrow \Sigma'$ is conservative when each Σ -model has a φ -expansion.*

For example a **FOL** signature morphism is conservative when all its components are injective

2.3. Comorphisms

In the literature there are several concepts of structure preserving mappings between institutions. The original one, introduced by [34], is adequate for expressing a ‘forgetful’ operation from a ‘more complex’ institution to a structurally ‘simpler’ one. However, the institution mapping which is appropriate for our task here of borrowing interpolation is that of institution comorphisms [37], previously know as ‘plain map’ in [48] or ‘representation’ in [67, 68]. Institution comorphisms realize the intuition of ‘embedding’ a ‘simpler’ institution into a ‘more complex’ one, which is dual to the intuition realized by the institution morphisms.

Definition 2.6 (Comorphisms). An institution comorphism $(\Phi, \alpha, \beta): \mathcal{I} \rightarrow \mathcal{I}'$ consists of

1. a functor $\Phi: \text{Sig} \rightarrow \text{Sig}'$,
2. a natural transformation $\alpha: \text{Sen} \Rightarrow \Phi; \text{Sen}',$ and
3. a natural transformation $\beta: \Phi^{\text{op}}; \text{Mod}' \Rightarrow \text{Mod}$

such that the following satisfaction condition holds

$$M' \models'_{\Phi(\Sigma)} \alpha_{\Sigma}(e) \text{ iff } \beta_{\Sigma}(M') \models_{\Sigma} e$$

for each signature $\Sigma \in |\text{Sig}|$, for each $\Phi(\Sigma)$ -model M' , and each Σ -sentence e .

The reader is invited to develop and check all details of the following three examples of comorphisms.

Example 2.8. The canonical embedding of **UNIV** into **FOL** can be expressed as a comorphism $(\Phi, \alpha, \beta): \text{UNIV} \rightarrow \text{FOL}$ such that Φ is the identity functor, α regards any universal sentence as a **FOL** sentence, and $\beta_{(S,F,P)}$ are also identities for each signature (S, F, P) . Similarly we have the embeddings **HCL** \rightarrow **UNIV** and **HCL** \rightarrow **FOL** as institution comorphisms.

Example 2.9. Let the institution of *first order equational logic*, denoted **FOEQL**, be the sub-institution of **FOL** determined by the algebraic signatures. This can be expressed as a comorphism $(\Phi, \alpha, \beta): \text{FOEQL} \rightarrow \text{FOL}$ such that $\Phi(S, F) = (S, F, \emptyset)$ for each algebraic signature (S, F) , $\alpha_{(S,F)}$ are identities, and $\beta_{(S,F)}$ are the obvious isomorphisms $\text{Mod}^{\text{FOL}}(S, F, \emptyset) \cong \text{Mod}^{\text{FOEQL}}(S, F)$.

Example 2.10. This example giving an embedding of first order equational logic into higher order logic is rather more sophisticated than the couple of examples above. We define a comorphism $(\Phi, \alpha, \beta): \text{FOEQL} \rightarrow \text{HNK}$ such that

- $\Phi(S, F) = (S, \overline{F})$ where $\overline{F}_{s_1 \rightarrow (s_2 \rightarrow \dots (s_n \rightarrow s) \dots)} = F_{s_1 \dots s_n \rightarrow s}$ for any sorts $s, s_1, \dots, s_n \in S$, and $F_x = \emptyset$ otherwise.
- The sentence translation α is determined by the following mapping, also denoted α , on terms: each term $t = \sigma(t_1, \dots, t_n)$ is assigned the Polish prefix translation of the term $\sigma(\alpha(t_1))(\alpha(t_2)) \dots (\alpha(t_n))$.
- For any **FOEQL** signature (S, F) we define $\beta_{(S,F)}: \text{Mod}^{\text{HNK}}(S, \overline{F}) \rightarrow \text{Mod}^{\text{FOEQL}}(S, F)$ by $\beta(M)_s = M_s$ for any each sort $s \in S$, and $\beta(M)_{\sigma(x_1, \dots, x_n)} = M_{\sigma(x_1)(x_2) \dots (x_n)}$ for each $\sigma \in F_{s_1 s_2 \dots s_n \rightarrow s}$. Note that $\beta_{(S,F)}$ are isomorphisms.

Definition 2.7. A comorphism $(\Phi, \alpha, \beta): \mathcal{I} \rightarrow \mathcal{I}'$ is conservative when for each signature Σ of \mathcal{I} , for each Σ -model M there exists a $\Phi(\Sigma)$ -model M' such that $M = \beta_{\Sigma}(M')$.

Note that all examples of comorphisms presented above are trivially conservative.

2.4. Presentations

Although comorphisms generally express an embedding relationship between institutions, they can also be used for ‘encoding’ a ‘more complex’ institution \mathcal{I} into a ‘simpler’ one \mathcal{I}' . The latter are especially useful for the borrowing methods, including our interpolation borrowing method developed here. In fact, the two of the applications of our main result will use ‘encoding’ comorphisms. In such encodings the structural complexity cost is shifted to the mapping Φ on the signatures, thus Φ maps signatures of \mathcal{I} to theories of \mathcal{I}' rather than signatures. In the following we give a general construction which facilitates the definition of ‘encoding’ comorphisms.

Definition 2.8 (Presentations). In any institution, a presentation is a pair (Σ, E) consisting of a signature Σ and a set E of Σ -sentences. A presentation morphism $\varphi: (\Sigma, E) \rightarrow (\Sigma', E')$ is a signature morphism $\varphi: \Sigma \rightarrow \Sigma'$ such that $E' \models \varphi(E)$.

Fact 2.1. Presentation morphisms are closed under the composition given by the composition of the signature morphisms.

This fact opens the door for the general construction given by the following definition.

Definition 2.9 (The institution of the presentations). Let $\mathcal{I} = (\text{Sig}, \text{Sen}, \text{Mod}, \models)$ be any institution. The institution of the presentations of \mathcal{I} , denoted by $\mathcal{I}^{\text{pres}} = (\text{Sig}^{\text{pres}}, \text{Sen}^{\text{pres}}, \text{Mod}^{\text{pres}}, \models^{\text{pres}})$, is defined by

- Sig^{pres} is the category $\mathbb{P}\text{res}$ of presentations of \mathcal{I} ,
- $\text{Sen}^{\text{pres}}(\Sigma, E) = \text{Sen}(\Sigma)$,
- $\text{Mod}^{\text{pres}}(\Sigma, E)$ is the full subcategory of $\text{Mod}(\Sigma)$ of those models which satisfy E , and
- for each (Σ, E) -model M and Σ -sentence e , $M \models_{(\Sigma, E)}^{\text{pres}} e$ if and only if $M \models_{\Sigma} e$.

Fact 2.2. $\mathcal{I}^{\text{pres}}$ is indeed an institution for any institution \mathcal{I} .

3. Borrowing interpolation: main theorem

This section is devoted to the main result of this paper. It is structured as follows:

1. We recall the concept of Craig interpolation at the level of abstract institutions.
2. We formulate and prove the generic borrowing interpolation theorem.

3.1. Craig interpolation

In the algebraic specification literature there are several institution independent formulations of Craig interpolation (which we abbreviate by *CI*). For example [64] is one of the first work introducing the concept of interpolation at the level of abstract institutions. The common feature of these formulations is that they generalise the conventional intersection-union (of signatures) framework to commuting squares of signature morphisms. In most cases these are required to be pushouts (like in [7, 8, 31, 65]) and when this is not the case the signature morphisms are required to be (abstract) inclusions [27].

It has been noticed in [20] that the mere formulation of CI does not require any extra technical assumptions besides a commuting square of signature morphisms, the role of such additional assumptions having more to do with the proof of CI rather than with its formulation.

Definition 3.1. A commuting square of signature morphisms

$$\begin{array}{ccc} \Sigma & \xrightarrow{\varphi_1} & \Sigma_1 \\ \varphi_2 \downarrow & & \downarrow \theta_1 \\ \Sigma_2 & \xrightarrow{\theta_2} & \Sigma' \end{array}$$

is a Craig interpolation square if and only if for each set E_1 of Σ_1 -sentences and set E_2 of Σ_2 -sentences such that $\theta_1(E_1) \models \theta_2(E_2)$ there exists a set E of Σ -sentences such that $E_1 \models \varphi_1(E)$ and $\varphi_2(E) \models E_2$. The set E is called the interpolant of E_1 and E_2 .

The works [60] and [27] argue successfully that the above formulation of CI in terms of sets of sentences is more natural than the more traditional formulations of CI in terms of single sentences. First, the applications of interpolation do not require the single sentence formulation. Secondly, in the traditional works on or using interpolation these two formulations are equivalent as shown by the following simple result.

Fact 3.1. *In a compact institution, if E_2 of Definition 3.1 is finite, then the interpolant E can be chosen to be finite too. Consequently, in compact institutions with finite conjunctions, the sets of sentence formulation of CI implies the single sentences formulation.*

Finally, perhaps the most important argument is that some institutions, not having conjunctions, lack CI in the single sentences version, but have it in the sets of sentences formulation. Typical examples for this situation are equational logic [20, 58–60] and Horn clause logic (**HCL**) [20, 58]. Moreover, a more careful look into this situation would reveal that in general the applications of interpolation require the set of sentences formulation. For example, this is clearly the case in the formal specification applications of interpolation.

Proposition 3.1. *For any CI square of signature morphisms*

$$\begin{array}{ccc} \Sigma & \xrightarrow{\varphi_1} & \Sigma_1 \\ \varphi_2 \downarrow & & \downarrow \theta_1 \\ \Sigma_2 & \xrightarrow{\theta_2} & \Sigma'' \end{array}$$

and any conservative signature morphism $\psi: \Sigma'' \rightarrow \Sigma'$, the commuting square

$$\begin{array}{ccc} \Sigma & \xrightarrow{\varphi_1} & \Sigma_1 \\ \varphi_2 \downarrow & & \downarrow \theta_1; \psi \\ \Sigma_2 & \xrightarrow{\theta_2; \psi} & \Sigma' \end{array}$$

is still a CI square.

Proof. Let $E_1 \subseteq \text{Sen}(\Sigma_1)$ and $E_2 \subseteq \text{Sen}(\Sigma_2)$ such that $(\theta_1; \psi)(E_1) \models (\theta_2; \psi)(E_2)$. Let us show that

$$\theta_1(E_1) \models \theta_2(E_2). \tag{1}$$

For this we consider any Σ'' -model M'' such that $M'' \models \theta_1(E_1)$. By the conservativeness of ψ there exists a Σ' -model M' such that $M' \upharpoonright_{\psi} = M''$. By the Satisfaction Condition $M' \models \psi(\theta_1(E_1))$, hence $M' \models \psi(\theta_2(E_2))$. By the Satisfaction Condition again it follows that $M'' = M' \upharpoonright_{\psi} \models \theta_2(E_2)$. Now by the CI square assumption, from (1) we obtain that there exists $E \subseteq \text{Sen}(\Sigma)$ such that $E_1 \models \varphi_1(E)$ and $\varphi_2(E) \models E_2$. \square

In principle, in the actual examples, CI is expected for pushout squares of signature morphisms, however in many situations only some pushout squares satisfy it. This intuition has been formulated first time in [8]. The interpolation concept below was introduced in [21] and is slightly simpler and more general than the so-called $(\mathcal{D}, \mathcal{T})$ -interpolation of [8].

Definition 3.2 ((\mathcal{L}, \mathcal{R})-interpolation). For any classes of signature morphisms $\mathcal{L}, \mathcal{R} \subseteq \text{Sig}$ in any institution, we say that the institution has the Craig (\mathcal{L}, \mathcal{R})-interpolation if each pushout square of signature morphisms of the form

$$\begin{array}{ccc} \bullet & \xrightarrow{\mathcal{L}} & \bullet \\ \mathcal{R} \downarrow & & \downarrow \\ \bullet & \longrightarrow & \bullet \end{array}$$

is a CI square.

Example 3.1. According to [39], **FOL** has Craig ($(i **), \text{Sig}^{\text{FOL}}$) and ($\text{Sig}^{\text{FOL}}, (i **)$)-interpolation where $(i **)$ is the class of all sort injective signature morphisms.

Example 3.2. According to [39], **PA** has Craig ($(i **), \text{Sig}^{\text{PA}}$) and ($\text{Sig}^{\text{PA}}, (i **)$)-interpolation. This result will also be obtained in the Corollary 4.1 below.

Example 3.3. According to Corollary 4.2 below, **POA** has Craig ($(i*), \text{Sig}^{\text{PA}}$) and ($\text{Sig}^{\text{PA}}, (i*)$)-interpolation.

Example 3.4.

1. According to [20, 58], **UNIV** and **HCL** have Craig ($\text{Sig}^{\text{FOL}}, (iii)$)-interpolation where (iii) is the class of injective signature morphisms (for **UNIV** this result is also obtained via Corollary 5.2 below), and
2. according to [58], **UNIV** and **HCL** have Craig ($(ie*), \text{Sig}^{\text{FOL}}$)-interpolation where $(ie*)$ is the class of the signature morphisms φ which are injective on the sorts and no ‘new’ operation symbol, i.e. outside the image of φ , is allowed to have the sort in the image of φ (i.e. if $\varphi: (S, F, P) \rightarrow (S', F', P')$ and $\sigma' \in F'_{w' \rightarrow s'}$ with $s' \in \varphi(s)$ then there exists $\sigma \in F_{w \rightarrow s}$ such that $\varphi(\sigma) = \sigma'$.) (For **UNIV** this result is also obtained via Corollary 5.1 below.)

The paper [58] gives a big list of such interpolation properties for various axiomatizable fragments of **FOL**. In [20] and [23] there are general interpolation theorems for abstract ‘Birkhoff’ institutions, i.e. institutions satisfying Birkhoff-style axiomatizability properties.

Example 3.5. According to Corollary 4.3 below, **HPOA** has Craig ($(ie), \text{Sig}^{\text{FOL}}$)-interpolation.

Example 3.6. According to Corollary 6.1 below, **HNK** has Craig ($(i*), \text{Sig}^{\text{HNK}}$) and ($\text{Sig}^{\text{HNK}}, (i*)$)-interpolation.

3.2. The borrowing theorem

The method of borrowing interpolation properties along institution comorphisms requires some special interpolation properties of the comorphism. These have been introduced in [21] and are recalled by the Definitions 3.3 and 3.4 below.

Definition 3.3 (Left interpolation property for comorphisms). For a fixed class $\mathcal{S} \subseteq \text{Sig}$ of signature morphisms, we say that an institution comorphism $(\Phi, \alpha, \beta): \mathcal{I} \rightarrow \mathcal{I}'$ has the Craig \mathcal{S} -left Interpolation property when for each $(\varphi_1: \Sigma \rightarrow \Sigma_1) \in \mathcal{S}$, for each set E_1 of Σ_1 -sentences and each set E_2 of $\Phi(\Sigma_1)$ -sentences such that $\alpha_{\Sigma_1}(E_1) \models' \Phi(\varphi_1)(E_2)$, there exists a set of Σ -sentences E such that $E_1 \models \varphi_1(E)$ and $\alpha_{\Sigma}(E) \models' E_2$.

$$\begin{array}{ccc} \text{Sen}(\Sigma) & \xrightarrow{\text{Sen}(\varphi_1)} & \text{Sen}(\Sigma_1) \\ \alpha_{\Sigma} \downarrow & & \downarrow \alpha_{\Sigma_1} \\ \text{Sen}'(\Phi(\Sigma)) & \xrightarrow{\text{Sen}'(\Phi(\varphi_1))} & \text{Sen}'(\Phi(\Sigma_1)) \end{array}$$

The following is the reflection in the mirror of the left property.

Definition 3.4 (Right interpolation property for comorphisms). For a fixed class $\mathcal{S} \subseteq \text{Sig}$ of signature morphisms, we say that an institution comorphism $(\Phi, \alpha, \beta): \mathcal{I} \rightarrow \mathcal{I}'$ has the Craig \mathcal{S} -right Interpolation property when for each $(\varphi_2: \Sigma \rightarrow \Sigma_2) \in \mathcal{S}$, for each set E_1 of $\Phi(\Sigma)$ -sentences and each set E_2 of Σ_2 -sentences such that $(\Phi(\varphi_2)(E_1) \models' \alpha_{\Sigma_2}(E_2))$, there exists a set of Σ -sentences E such that $E_1 \models \alpha_{\Sigma}(E)$ and $\varphi(E) \models' E_2$.

$$\begin{array}{ccc} \text{Sen}(\Sigma) & \xrightarrow{\alpha_{\Sigma}} & \text{Sen}'(\Phi(\Sigma)) \\ \text{Sen}(\varphi_2) \downarrow & & \downarrow \text{Sen}'(\Phi(\varphi_2)) \\ \text{Sen}(\Sigma_2) & \xrightarrow{\alpha_{\Sigma_2}} & \text{Sen}'(\Phi(\Sigma_2)) \end{array}$$

We also need the following relaxed variant of the pushout concept.

Definition 3.5 (Quasi-pushouts). In any institution with pushouts of signatures, a commuting square of signatures

$$\begin{array}{ccc} \Sigma & \xrightarrow{\varphi_1} & \Sigma_1 \\ \varphi_2 \downarrow & & \downarrow \theta_1 \\ \Sigma_2 & \xrightarrow{\theta_2} & \Sigma' \end{array}$$

is a quasi-pushout when the (unique) signature morphism $\psi: \Sigma'' \rightarrow \Sigma'$ from the vertex Σ'' of the pushout of φ_1 and φ_2 to Σ' is conservative.

Theorem 3.1 (Borrowing interpolation). Let $(\Phi, \alpha, \beta): \mathcal{I} \rightarrow \mathcal{I}'$ be a conservative institution comorphism such that Φ maps pushouts to quasi-pushouts, and let $\mathcal{L}, \mathcal{R} \subseteq \text{Sig}$ be classes of signature morphisms such that \mathcal{I}' has the Craig $(\Phi(\mathcal{L}), \Phi(\mathcal{R}))$ -interpolation. Then \mathcal{I} has Craig $(\mathcal{L}, \mathcal{R})$ -interpolation.

Proof. Consider a pushout of signature morphisms

$$\begin{array}{ccc} \Sigma & \xrightarrow{\varphi_1} & \Sigma_1 \\ \varphi_2 \downarrow & & \downarrow \theta_1 \\ \Sigma_2 & \xrightarrow{\theta_2} & \Sigma' \end{array}$$

such that $\varphi_1 \in \mathcal{L}$ and $\varphi_2 \in \mathcal{R}$ and $E_1 \subseteq \text{Sen}(\Sigma_1)$ and $E_2 \subseteq \text{Sen}(\Sigma_2)$ such that $\theta_1(E_1) \models \theta_2(E_2)$.

The latter relation leads to $\alpha_{\Sigma'}(\theta_1(E_1)) \models' \alpha_{\Sigma'}(\theta_2(E_2))$ which by the naturality of α further leads us to the interpolation problem $\Phi(\theta_1)(\alpha_{\Sigma_1}(E_1)) \models' \Phi(\theta_2)(\alpha_{\Sigma_2}(E_2))$ for the following quasi-pushout square of signature morphisms in \mathcal{I}'

$$\begin{array}{ccc} \Phi(\Sigma) & \xrightarrow{\Phi(\varphi_1)} & \Phi(\Sigma_1) \\ \Phi(\varphi_2) \downarrow & & \downarrow \Phi(\theta_1) \\ \Phi(\Sigma_2) & \xrightarrow{\Phi(\theta_2)} & \Phi(\Sigma') \end{array}$$

By considering the pushout of $\Phi(\varphi_1)$ and $\Phi(\varphi_2)$, by Proposition 3.1, and by the Craig $(\Phi(\mathcal{L}), \Phi(\mathcal{R}))$ -interpolation property of \mathcal{I}' , we find $E_0 \subseteq \text{Sen}'(\Phi(\Sigma))$ such that

$$\alpha_{\Sigma_1}(E_1) \models \Phi(\varphi_1)(E_0) \text{ and } \Phi(\varphi_2)(E_0) \models \alpha_{\Sigma_2}(E_2).$$

Let us assume the Craig \mathcal{L} -left interpolation for the institution comorphism. Then we can find $E \subseteq \text{Sen}(\Sigma)$ such that

$$E_1 \models \varphi_1(E) \text{ and } \alpha_\Sigma(E) \models E_0.$$

By applying $\Phi(\varphi_2)$ to this we get that $\Phi(\varphi_2)(\alpha_\Sigma(E)) \models \Phi(\varphi_2)(E_0)$. By the naturality of α we get $\alpha_{\Sigma_2}(\varphi_2(E)) \models \Phi(\varphi_2)(E_0) \models \alpha_{\Sigma_2}(E_2)$. Finally, by the conservativeness of the institution comorphism this can be simplified to $\varphi_2(E) \models E_2$.

The case when the institution comorphism has \mathcal{R} -right interpolation is handled similarly to the \mathcal{L} -left interpolation case by getting $E \subseteq \text{Sen}(\Sigma)$ such that

$$E_0 \models \alpha_\Sigma(E) \text{ and } \varphi_2(E) \models E_2.$$

□

In all our applications but one, namely interpolation in higher order logic, the comorphisms involved preserve pushouts of signatures, therefore they involve the corresponding slightly less general variant of Theorem 3.1.

4. Interpolation in partial and preordered algebra

The main condition to be solved in the applications of Theorem 3.1 is that of the interpolation property for the institution comorphism. The following immediate result gives an easy sufficient solution for this condition and in principle can be applied to situations when the two institutions involved have the same expressive power.

Fact 4.1. *Any conservative institution comorphism $(\Phi, \alpha, \beta): \mathcal{I} \rightarrow \mathcal{I}'$ for which each α_Σ is surjective modulo semantical equivalence \models , has both the Craig $\text{Sig}^{\mathcal{I}}$ -left and right interpolation properties.*

For using the situation described by the Fact 4.1 to interpolation results from **FOL** to **PA**, we need to find an adequate comorphism **PA** \rightarrow **FOL**. But this is not possible because **PA** is too refined to be directly embedded into **FOL**. However a comorphism fulfilling the conditions required by Theorem 3.1 and by Fact 4.1 can be defined from **PA** to **FOL**^{pres}. And now we have run into another problem: we have to lift the interpolation property from **FOL** to **FOL**^{pres}. Fortunately this can be done uniformly at the level of abstract institutions but for a stronger form of interpolation, the so-called *Craig-Robinson interpolation*. In this section we will implement the ideas sketched above, in the following order.

1. We define the concept of Craig-Robinson interpolation.
2. We give a general result for lifting interpolation from a base institution \mathcal{I} to the institution of its presentation $\mathcal{I}^{\text{pres}}$.
3. We define a comorphism **PA** \rightarrow **FOL**^{pres} and will apply the general borrowing interpolation result given by Theorem 3.1 for this comorphism.
4. We replicate 3. above to the case of comorphisms **POA** \rightarrow **FOL**^{pres} and **HPOA** \rightarrow **HCL**^{pres} in order to derive interpolation results in **POA** and **HPOA**.

4.1. Craig-Robinson interpolation

The Craig interpolation property can be strengthened by adding to the ‘primary’ premises E_1 a set Γ_2 (of Σ_2 -sentences) as ‘secondary’ premises. Craig-Robinson interpolation plays an important role in specification language theory, see [5, 27, 31]. The name ‘Craig-Robinson’ interpolation has been used for instances of this property in [31, 63, 69] and ‘strong Craig interpolation’ has been used in [27].

Definition 4.1 (Craig-Robinson interpolation). *In any institution we say that a commuting square of signature morphisms*

$$\begin{array}{ccc} \Sigma & \xrightarrow{\varphi_1} & \Sigma_1 \\ \varphi_2 \downarrow & & \downarrow \theta_1 \\ \Sigma_2 & \xrightarrow{\theta_2} & \Sigma' \end{array}$$

is a Craig-Robinson Interpolation square (abbreviated CRI square) when for each set E_1 of Σ_1 -sentences and each sets E_2 and Γ_2 of Σ_2 -sentences, if $\theta_1(E_1) \cup \theta_2(\Gamma_2) \models_{\Sigma'} \theta_2(E_2)$, then there exists a set E of Σ -sentences such that $E_1 \models_{\Sigma_1} \varphi_1(E)$ and $\Gamma_2 \cup \varphi_2(E) \models_{\Sigma_2} E_2$.

Also the $(\mathcal{L}, \mathcal{R})$ -interpolation concept of Definition 3.2 can be extended in a straightforward way from Craig interpolation to Craig-Robinson interpolation.

By taking Γ_2 to be the empty set \emptyset we can see that

Fact 4.2. *Any CRI square is also a CI square.*

The opposite implication does not hold in general. The following gives a sufficient condition when CI and CRI are equivalent interpolation concepts.

Proposition 4.1. *In any compact institution that has implications, a commuting square of signature morphisms is a CRI square if and only if it is a CI square.*

Proof. We focus only on the non-trivial part, that CI implies CRI. Consider $E_1 \subseteq \text{Sen}(\Sigma_1)$ and $E_2, \Gamma_2 \subseteq \text{Sen}(\Sigma_2)$ such that $\theta_1(E_1) \cup \theta_2(\Gamma_2) \models \theta_2(E_2)$.

First we notice that without loss of generality we may assume that E_2 consists of only one sentence e , i.e. $E_2 = \{e\}$. Indeed, if we assumed that CRI property holds for each $e \in E_2$, let E_e be the interpolant corresponding to each $e \in E_2$. Then $\bigcup_{e \in E_2} E_e$ is an interpolant corresponding to E_2 .

Because we may assume that $E_2 = \{e\}$, then by the quasi-compactness assumption, we may assume without loss of generality that E_1 and Γ_2 are finite.

Let $\Gamma_2 \Rightarrow e$ denote $\gamma_1 \Rightarrow (\dots \Rightarrow (\gamma_n \Rightarrow e))$ where $\Gamma_2 = \{\gamma_1, \dots, \gamma_n\}$. Then we have that $\theta_1(E_1) \models \theta_2(\Gamma_2 \Rightarrow e)$. By CI there exists $E \subseteq \text{Sen}(\Sigma)$ such that $E_1 \models \varphi_1(E)$ and $\varphi_2(E) \models \Gamma_2 \Rightarrow e$. But the latter is equivalent to $\varphi_2(E) \cup \Gamma_2 \models e$. \square

Although one may get the feeling that CRI codes a form of implication and therefore it is expected only in institutions having semantic implications, it is not so. In [23] one can find examples of institutions without semantic implications having CRI for a wide class of pushout squares of signature morphisms.

4.2. Lifting interpolation to presentations

Proposition 4.2. *For any institution \mathcal{I} and a class $\mathcal{S} \subseteq \text{Sig}$ of signature morphisms let $\mathcal{S}^{\text{pres}}$ be the class of presentation morphisms φ such that $\varphi \in \mathcal{S}$ (as signature morphism). The institution $\mathcal{I}^{\text{pres}}$ of the presentations of \mathcal{I} has the Craig-Robinson $(\mathcal{L}^{\text{pres}}, \mathcal{R}^{\text{pres}})$ -interpolation if \mathcal{I} has the Craig-Robinson $(\mathcal{L}, \mathcal{R})$ -interpolation.*

Proof. Consider a pushout of presentations

$$\begin{array}{ccc} (\Sigma, \Gamma) & \xrightarrow{\varphi_1} & (\Sigma_1, \Gamma_1) \\ \varphi_2 \downarrow & & \downarrow \theta_1 \\ (\Sigma_2, \Gamma_2) & \xrightarrow{\theta_2} & (\Sigma', \Gamma') \end{array}$$

such that $\varphi_1 \in \mathcal{L}$, $\varphi_2 \in \mathcal{R}$ and let $E_1 \subseteq \text{Sen}(\Sigma_1)$ and $E_2, K_2 \in \text{Sen}(\Sigma_2)$ such that $\theta_1(E_1) \cup \theta_2(E_2) \models_{(\Sigma', \Gamma')} \theta_2(E_2)$.

By the general construction of colimits of presentations (see [34]) $\Gamma' \models \theta_1(\Gamma_1) \cup \theta_2(\Gamma_2)$ and $(\varphi_1, \varphi_2, \theta_1, \theta_2)$ form a pushout square in Sig . Therefore $\theta_1(E_1 \cup \Gamma_1) \cup \theta_2(K_2 \cup \Gamma_2) \models_{\Sigma'} \theta_2(E_2)$. By the CRI property for the base institution \mathcal{I} , there exists an interpolant $E \subseteq \text{Sen}(\Sigma)$ such that

$$E_1 \cup \Gamma_1 \models_{\Sigma_1} \varphi_1(E) \text{ and } \varphi_2(E) \cup K_2 \cup \Gamma_2 \models_{\Sigma_2} E_2$$

But these just mean

$$E_1 \models_{(\Sigma_1, \Gamma_1)} \varphi_1(E) \text{ and } \varphi_2(E) \cup K_2 \models_{(\Sigma_2, \Gamma_2)} E_2$$

which shows the CRI property for presentations. \square

4.3. Interpolation in PA

In the algebraic specification literature (see [52] for example) there are two major comorphisms $\mathbf{PA} \rightarrow \mathbf{FOL}^{\text{pres}}$. A third one has been recently developed in [24]. Two of them encodes the partial operations as total operations, and only one encodes them as relations. This latter comorphism, also used for borrowing definability in [57] is the one which is appropriate for our task here. We recall its definition below:

- Each \mathbf{PA} -signature (S, TF, PF) gets mapped to the \mathbf{FOL} presentation $((S, TF, \overline{PF}), \Gamma_{(S, TF, PF)})$ such that $\overline{PF}_{wS} = PF_{w \rightarrow S}$ for each $w \in S^*$ and $s \in S$, and

$$\Gamma_{(S, TF, PF)} = \{(\forall X \uplus \{y, z\})\sigma(X, y) \wedge \sigma(X, z) \Rightarrow (y = z) \mid \sigma \in PF\}$$

- Each (S, TF, \overline{PF}) -model M gets mapped to the partial (S, TF, PF) -algebra $\beta(M)$ such that
 - $\beta(M)_x = M_x$ for each $x \in S$ or $x \in TF$,
 - for each $\sigma \in PF$, if $(m, m_0) \in M_\sigma$ then $\beta(M)_\sigma(m) = m_0$.
- α commutes with the quantifiers and the Boolean connectives, and

$$\alpha(t \stackrel{e}{=} t') = (\exists X \uplus \{x_0\})\text{bind}(t, x_0) \wedge \text{bind}(t', x_0)$$

where for each (S, TF, PF) -term t and variable x , $bind(t, x)$ is a (finite) conjunction of atoms defined by

$$bind(\sigma(t_1 \dots t_n), x) = \bigwedge_{1 \leq i \leq n} bind(t_i, x_i) \wedge \begin{cases} \sigma(x_1, \dots, x_n) = x & \text{when } \sigma \in TF \\ \sigma(x_1, \dots, x_n, x) & \text{when } \sigma \in PF \end{cases}$$

and X is the set of the new constants introduced by $bind(t, x_0)$ and $bind(t', x_0)$.

Corollary 4.1. *PA has Craig $((i * *), \mathbb{S}ig^{\mathbf{PA}})$ and $(\mathbb{S}ig^{\mathbf{PA}}, (i * *))$ -interpolation.*

Proof. We apply Theorem 3.1 for the comorphism above and for the **FOL** interpolation properties presented in Section 3. We do this by means of Propositions 4.1 and 4.2 and of Fact 4.1. The conditions required are fulfilled as follows:

- That Φ maps pushouts of **PA** signatures to pushouts of **FOL** presentations can be seen immediately from the general result about colimits of presentations in abstract institutions [34].
- The comorphism is conservative because the β s are isomorphisms.
- α is surjective modulo \models because it preserves the quantifications and the Boolean connectives, and because it is surjective on the atoms ($\alpha(t \stackrel{e}{=} t') \models (t = t')$ for each equational (S, TF, \overline{PF}) -atom and $\alpha(\sigma(t_1, \dots, t_n) \stackrel{e}{=} t) \models \sigma(t_1, \dots, t_n, t)$ for each relational (S, TF, \overline{PF}) -atom.)

□

4.4. Interpolation in **POA**

There exists a comorphism $(\Phi, \alpha, \beta): \mathbf{POA} \rightarrow \mathbf{FOL}^{\text{pres}}$ such that

- each algebraic signature (S, F) gets mapped to the **FOL**-presentation $((S, F, \{\leq_s\}_{s \in S}), pre_{(S, F)})$ such that
 - for each sort symbol $s \in S$ the arity of \leq_s is ss , and
 - $pre_{(S, F)}$ contains the preorder axioms for each \leq_s and all axioms stating the preorder functoriality of the operations of F .
- α extends canonically the mapping $\alpha(t \leq t') = (t \leq_s t')$ (for s being the sort of t and t'), and
- β s are identities.

By using this comorphism we have a proof of the following which is similar to the proof of Corollary 4.1.

Corollary 4.2. *POA has Craig $((i *), \mathbb{S}ig^{\mathbf{POA}})$ and $(\mathbb{S}ig^{\mathbf{POA}}, (i *))$ -interpolation.*

4.5. Interpolation in **HPOA**

The comorphism $\mathbf{POA} \rightarrow \mathbf{FOL}^{\text{pres}}$ above when restricted to **HPOA** appears as a comorphism $\mathbf{HPOA} \rightarrow \mathbf{HCL}^{\text{pres}}$. This is because the axioms $\text{pre}_{(S,F)}$ are all Horn clauses and moreover α trivially maps **POA** Horn clauses to **FOL** Horn clauses. This enables the following result.

Corollary 4.3. ***HPOA** has Craig $((ie), \text{Sig}^{\mathbf{POA}})$ -interpolation.*

Proof. According to [23], **HCL** has Craig-Robinson $((ie^*), \text{Sig}^{\mathbf{FOL}})$ -interpolation. By Proposition 4.2 this lifts to $\mathbf{HCL}^{\text{pres}}$. As CI is less general la CRI, it follows that $\mathbf{HCL}^{\text{pres}}$ has Craig $((ie^*), \text{Sig}^{\mathbf{FOL}})$ -interpolation. Since the comorphism $\mathbf{HPOA} \rightarrow \mathbf{HCL}^{\text{pres}}$ maps (ie) -morphisms of signatures to (ie^*) -morphisms of signatures, and it is bijective both on α and β components, the conclusion follows by virtue of Theorem 3.1 through Fact 4.1. \square

Note that the kind of argument as in Corollary 4.3 cannot be applied to the Horn clause fragment of **PA** because the comorphism $\mathbf{PA} \rightarrow \mathbf{FOL}^{\text{pres}}$ above does not allow a restriction and co-restriction to the corresponding Horn clause fragments, this unfortunate situation being due to α rather than to Φ . Moreover, none of the other two comorphisms $\mathbf{PA} \rightarrow \mathbf{FOL}^{\text{pres}}$ of the literature would do it.

5. Interpolation for universal sentences

In this section we derive the interpolation properties for **UNIV**, the sub-institution of **FOL** of the universal sentences, which have been mentioned in Section 3. This example falls into the category of the applications of Theorem 3.1 in which the institution \mathcal{I} has less expressive power than \mathcal{I}' . In this example the institutions **UNIV** and **FOL** share the same signatures and models but the sentences of **UNIV** are much less than the sentences of **FOL**. Obviously, in such situations there is no hope to use the method employed in the previous Section 4 since α 's are inclusions rather than surjections. This means that establishing the left/right interpolation properties for the comorphism requires more refined work. In fact this will be the main technical task to be solved in this section. For this section we assume familiarity with the concept of ultraproduct of models in **FOL** (see [11] for the classic approach to ultraproducts and [18, 23, 28] for an institution-independent approach).

Proposition 5.1. *The institution (embedding) comorphism $\mathbf{UNIV} \rightarrow \mathbf{FOL}$ has the Craig (ie^*) -left interpolation property.*

Proof. Consider $\varphi_1 : \Sigma \rightarrow \Sigma_1$ be a (ie^*) morphism of signatures and let E_1 be a set of universal Σ_1 -sentences and E_2 be set of **FOL** Σ -sentences such that $E_1 \models \varphi_1(E_2)$. In the following the operators $(-)^*$ on classes of models are those of the institution **UNIV**. We define the interpolant $E = \varphi_1^{-1}(E_1^{**})$. Since $E_1 \models \varphi_1(E)$ is immediate we have only to prove that $E \models E_2$.

Let M_2 be a model such that $M_2 \models E$. We have that $M_2 \models \varphi_1^{-1}(E_1^{**})$ which means $M_2 \in (\varphi_1^{-1}(E_1^{**}))^*$. By the satisfaction condition $(\varphi_1^{-1}(E_1^{**}))^* = (E_1^* \upharpoonright_{\varphi_1})^{**}$, hence $M_2 \in (E_1^* \upharpoonright_{\varphi_1})^{**}$. By the well known axiomatizability property for universal sentences (see [2]) we have that there exists an ultraproduct M of models from $E_1^* \upharpoonright_{\varphi_1}$ and a closed injective model homomorphism $h : M_2 \rightarrow M$. Because in **FOL** all model reducts preserve filtered products,³ we may find a φ_1 -expansion N of M which is an ultraproduct of models from E_1^* . By the Fundamental Ultraproducts Theorem (see [11, 23]), $N \models E_1$. Because φ_1 is an (ie^*) morphism of signatures, there exists a φ_1 -expansion $h' : N_2 \rightarrow N$ of h (which also implies that N_2 is a φ_1 -expansion of M_2) which is defined as follows:

³A proof of this rather basic fact can be found for example in [23].

- $(N_2)_{\varphi_1(s)} = (M_2)_s$ and $h'_{\varphi_1(s)} = h_s$ for each sort s of Σ ,
- $(N_2)_{s'} = N_{s'}$ and $h'_{s'} = 1_{N_{s'}}$ for each sort s' in Σ_1 outside the image of φ_1 ,
- $(N_2)_{\sigma'}(x) = h'^{-1}(N_{\sigma'}(h'(x)))$ for each operation symbol σ' in Σ_1 and for each appropriate list x of arguments. This is correctly defined because of the encapsulation condition e on the operations, because of the injectivity of h' , and furthermore it makes h' a homomorphism for the operations, and
- for each relation π' in Σ_1 we define $(N_2)_{\pi'} = h'^{-1}(N_{\pi'})$.

Note that h' is also closed. Because universal sentences are preserved by closed submodels and because $N \models E_1$ we have that $N_2 \models E_1$ too. Since $E_1 \models \varphi_1(E_2)$ we have that $N_2 \models \varphi_1(E_2)$. By the satisfaction condition this means $M_2 \models E_2$. \square

By Theorem 3.1 we may now ‘borrow’ one of the interpolation properties of **FOL** presented in Section 3, namely that **FOL** has Craig $((i * *), \text{Sig}^{\text{FOL}})$ -interpolation.

Corollary 5.1. *UNIV has Craig $((ie*), \text{Sig})$ -interpolation.*

Proposition 5.2. *The institution (embedding) comorphism $\text{UNIV} \rightarrow \text{FOL}$ has the Craig (iii)-right interpolation property.*

Proof. Let $\varphi_2: \Sigma \rightarrow \Sigma_2$ be an (iii) morphism of signatures and let E_1 be a set of **FOL** Σ -sentences and E_2 be a set of universal Σ_2 -sentences such that $\varphi_2(E_1) \models E_2$. Like in the proof of Proposition 5.1, in the following the operators $(-)^*$ on classes of models are those of the institution **UNIV**. The interpolant is defined as $E = E_1^{**}$. Obviously $E_1 \models E$. We have to prove that $\varphi_2(E) \models E_2$.

Consider a model M_2 such that $M_2 \models \varphi_2(E)$. By the satisfaction condition this means that $M_2 \upharpoonright_{\varphi_2} \models E$ which can be written as $M_2 \upharpoonright_{\varphi_2} \in (E_1^{**})^* = (E_1^*)^{**}$. By the same axiomatizability result invoked in the proof of Proposition 5.1, there exists a closed injective model homomorphism $h: M_2 \upharpoonright_{\varphi_2} \rightarrow N$ where N is an ultraproduct of models from E_1^* . By the Fundamental Ultraproducts Theorem we have that $N \in E_1^*$ too. Because φ_2 is an (iii) morphism of signatures, there exists a φ_2 -expansion $h': M_2 \rightarrow N_2$ of h defined as follows:

- $(N_2)_{\varphi_2(z)} = N_z$ for each z sort, or operation symbol, or relation symbol, of Σ , and $h'_{\varphi_2(s)} = h_s$ for each sort s of Σ ; this is correctly defined because φ_2 is injective,
- $(N_2)_{s'} = (M_2)_{s'}$ and $h'_{s'} = 1_{(M_2)_{s'}}$ for each sort s' in Σ_2 outside the image of φ_2 ,
- for each operation symbol $\sigma' \in \Sigma_2$ outside the image of φ_2 , with arity w and sort s
 - $(N_2)_{\sigma'}(h'_w(x)) = h'_s((M_2)_{\sigma'}(x))$ if $x \in (M_2)_w$,
 - otherwise let $(M_2)_{\sigma'}(x)$ be any element of $(N_2)_s$.
- for each relation symbol $\pi' \in \Sigma_2$, outside the image of φ_2 , with arity w , $(N_2)_{\pi'} = h'_w((M_2)_{\pi'})$.

Note that h' is closed too. By the satisfaction condition $N_2 \models \varphi_2(E_1)$. Because $\varphi_2(E_1) \models E_2$ we have that $N_2 \models E_2$. But E_2 are universal, thus they are preserved by closed submodels, hence $M_2 \models E_2$. \square

By Theorem 3.1 we may now ‘borrow’ the other interpolation property of **FOL** presented in Section 3, namely that **FOL** has Craig $(\text{Sig}^{\text{FOL}}, (i * *))$ -interpolation.

Corollary 5.2. *UNIV has Craig $(\text{Sig}, (iii))$ -interpolation.*

6. Interpolation in higher order logic

Because of its higher order type structure, higher order logic with Henkin semantics **HNK** is significantly more refined in terms of expressive power than **FOL**. This claim is also supported by the example of the embedding of **FOEQL** into **HNK** presented in Section 2. In this section we ‘borrow’ interpolation properties from **FOL** to **HNK** in a manner similar to that of Section 4, by addressing some special problems raised by this example. For this we have to do the following:

1. We define an institution comorphism $\mathbf{HNK} \rightarrow \mathbf{FOEQL}^{\text{pres}}$.
2. We apply the borrowing Theorem 3.1.

6.1. Encoding higher order logic into first order logic

The following comorphism $(\Phi, \alpha, \beta) : \mathbf{HNK} \rightarrow \mathbf{FOEQL}^{\text{pres}}$ is based upon ideas from [51]:

- Each **HNK**-signature (S, F) gets mapped to the presentation $((\vec{S}, \vec{F}), \Gamma_{(S,F)})$ where
 - \vec{S} is the set of all S -types,
 - $\vec{F}_s = F_s$ for each $s \in \vec{S}$, $\vec{F}_{[(s \rightarrow s')_s] \rightarrow s'} = \{\text{app}_{s,s'}\}$ for all $s, s' \in \vec{S}$ and $\vec{F}_{w \rightarrow s} = \emptyset$ otherwise.
 - $\Gamma_{(S,F)} = \{(\forall f)(\forall g)((\forall x)\text{app}_{s,s'}(f, x) = \text{app}_{s,s'}(g, x)) \Rightarrow (f = g) \mid s, s' \in \vec{S}\}$
 - $\beta_{(S,F)}(M) = \bar{M}$ where \bar{M} is the following inductively (on the structure of the types) defined **HNK**-model:
 - For each $s \in S$, $\bar{M}_s = M_s$; let us denote this identity by fun_s^M .
 - For each type $s \rightarrow s' \in \vec{S}$, assuming that fun_s^M and $\text{fun}_{s'}^M$ have been defined and are bijective, let us denote by $\text{fun}_{s \rightarrow s'}^M$ the canonical bijection $[M_s \rightarrow M_{s'}] \rightarrow [\bar{M}_s \rightarrow \bar{M}_{s'}]$ defined by

$$\text{fun}_{s \rightarrow s'}^M(f)(x) = \text{fun}_{s'}^M(f((\text{fun}_s^M)^{-1}(x))).$$
 We define $\bar{M}_{s \rightarrow s'} = \text{fun}_{s \rightarrow s'}^M(M_{s \rightarrow s'}^\square)$ where $M_{s \rightarrow s'}^\square$ is the image of $M_{s \rightarrow s'}$ through the canonical injection $M_{s \rightarrow s'} \rightarrow [M_s \rightarrow M_{s'}]$.
- Note that all these determine also an isomorphism $\text{fun}^M : M \rightarrow \bar{M}$ (here \bar{M} is canonically regarded as a **FOEQL** $((\vec{S}, \vec{F}), \Gamma_{(S,F)})$ -model with **app** interpreted as ordinary functional application).
- α is defined as the canonical extension of the mapping on the terms α^{tm} defined by $\alpha^{\text{tm}}(tt') = \text{app}(\alpha^{\text{tm}}(t), \alpha^{\text{tm}}(t'))$.

The reader may complete the details of this definition (such as the definitions of Φ on the signature morphisms and of the $\beta_{(S,F)}$ on the model homomorphisms) by herself/himself.

Fact 6.1. $\beta_{(S,F)}$ is an equivalence of categories with an ‘inverse’ $\bar{\beta}_{(S,F)}$ such that $\bar{\beta}_{(S,F)}; \beta_{(S,F)} = 1$ and $\text{fun} : 1 \xrightarrow{\cong} \beta_{(S,F)}; \bar{\beta}_{(S,F)}$ is isomorphism.

We also leave to the reader the task to check that (Φ, α, β) is indeed a comorphism $\mathbf{HNK} \rightarrow \mathbf{FOEQL}^{\text{pres}}$.

6.2. Borrowing interpolation along comorphism $\mathbf{HNK} \rightarrow \mathbf{FOEQL}^{\text{pres}}$

This example has the following interesting feature that distinguishes it from the other examples and which calls for an application of Theorem 3.1 in its full generality: the signature component Φ of the comorphism $\mathbf{HNK} \rightarrow \mathbf{FOEQL}^{\text{pres}}$ defined above does not preserve pushouts in general, but rather maps pushouts to quasi-pushouts. This is due to the way pushouts of \mathbf{HNK} signature morphisms do exist, as shown in the following.

Proposition 6.1. [14] *For any \mathbf{HNK} signature morphisms $\varphi_1: (S, F) \rightarrow (S_1, F_1)$ and $\varphi_2: (S, F) \rightarrow (S_2, F_2)$ their pushout exists and is constructed as follows:*

1. We consider the following pushout in \mathbf{Set} :

$$\begin{array}{ccc} S & \xrightarrow{\varphi_1^{\text{st}}} & S_1 \\ \varphi_2^{\text{st}} \downarrow & & \downarrow \theta_1^{\text{st}} \\ S_2 & \xrightarrow{\theta_2^{\text{st}}} & S' \end{array}$$

2. We extend the above pushout to a commuting diagram of sets of types:

$$\begin{array}{ccc} \vec{S} & \xrightarrow{\varphi_1^{\text{type}}} & \vec{S}_1 \\ \varphi_2^{\text{type}} \downarrow & & \downarrow \theta_1^{\text{type}} \\ \vec{S}_2 & \xrightarrow{\theta_2^{\text{type}}} & \vec{S}' \end{array}$$

Note that this is not the pushout of φ_1^{type} and φ_2^{type} , their pushout being given by the subset S'' of \vec{S}' defined by $S'' = \theta_1^{\text{type}}(\vec{S}_1) \cup \theta_2^{\text{type}}(\vec{S}_2)$.

3. For each type $s' \in \vec{S}'$, if $s' \notin S''$ then we let $F'_{s'} = \emptyset$, otherwise $F'_{s'}$ is defined by the following pushout of sets of operation symbols:

$$\begin{array}{ccc} \uplus_{\theta(\varphi(s))} F_s & \longrightarrow & \uplus_{\theta_1(s_1)=s'} (F_1)_{s_1} \\ \downarrow & & \downarrow \\ \uplus_{\theta_2(s_2)=s'} (F_2)_{s_2} & \longrightarrow & F'_{s'} \end{array}$$

Then

$$\begin{array}{ccc} (S, F) & \xrightarrow{\varphi_1} & (S_1, F_1) \\ \varphi_2 \downarrow & & \downarrow \theta_1 \\ (S_2, F_2) & \xrightarrow{\theta_2} & (S', F') \end{array}$$

is a pushout of \mathbf{HNK} signature morphisms.

That Φ does not map \mathbf{HNK} signature pushouts to pushouts of presentations in \mathbf{FOEQL} can be seen easily from the remark in the second item of Proposition 6.1. However Φ does map pushouts to quasi-pushouts.

Proposition 6.2. *The signature translation functor Φ of the comorphism $\mathbf{HNK} \rightarrow \mathbf{FOEQL}^{\text{pres}}$ maps pushouts to quasi-pushouts.*

Proof. Consider a pushout (θ_1, θ_2) of a span (φ_1, φ_2) of \mathbf{HNK} signature morphisms and let (χ_1, χ_2) be the pushout of $(\Phi(\varphi_1), \Phi(\varphi_2))$.

$$\begin{array}{ccc}
(S, F) \xrightarrow{\varphi_1} (S_1, F_1) & & ((\vec{S}, \vec{F}), \Gamma_{(S,F)}) \xrightarrow{\Phi(\varphi_1)} ((\vec{S}_1, \vec{F}_1), \Gamma_{(S_1,F_1)}) \\
\varphi_2 \downarrow & & \Phi(\varphi_2) \downarrow \\
(S_2, F_2) \xrightarrow{\theta_2} (S', F') & & ((\vec{S}_2, \vec{F}_2), \Gamma_{(S_2,F_2)}) \xrightarrow{\chi_2} ((S'', F''), \Gamma'') \\
& & \downarrow \chi_1 \\
& & ((\vec{S}', \vec{F}'), \Gamma_{(S',F')}) \\
& & \uparrow \psi \\
& & ((\vec{S}, \vec{F}), \Gamma_{(S',F')})
\end{array}$$

Let $\psi: ((S'', F''), \Gamma'') \rightarrow ((\vec{S}', \vec{F}'), \Gamma_{(S',F')})$ be the unique \mathbf{FOEQL} presentation morphism such that $\chi_k; \psi = \Phi(\theta_k)$ for $k \in \{1, 2\}$. By Proposition 6.1 we have that ψ is an inclusion since $S'' \subseteq \vec{S}'$ and \vec{F}' just adds to F'' the set $\{\text{app}_{s,s'} \mid s \rightarrow s' \notin S''\}$. Therefore any model M'' of $((S'', F''), \Gamma'')$ has a ψ -expansion which interprets any $s \in \vec{S}' \setminus S''$ and any $\text{app}_{s,s'}$ for $s \rightarrow s' \notin S''$ as empty. \square

Let us note the following:

Fact 6.2. *For each \mathbf{HNK} signature (S, F) , the sentence translation $\alpha_{(S,F)}$ is a bijection.*

Now we have established all the necessary conditions for the application of Theorem 3.1 for the comorphism $\mathbf{HNK} \rightarrow \mathbf{FOEQL}^{\text{pres}}$ and for the interpolation properties of \mathbf{FOEQL} (which are those of \mathbf{FOL} simplified for the situation given by the absence of the relation symbols). Doing this by means of Propositions 4.1 and 4.2 and of Fact 4.1 in the manner of the borrowing of interpolation properties from \mathbf{FOL} to \mathbf{PA} done in Section 4 leads to the following interpolation properties in \mathbf{HNK} :

Corollary 6.1. *\mathbf{HNK} has Craig $((i^*), \text{Sig})$ and $(\text{Sig}, (i^*))$ -interpolation.*

Acknowledgement. I am grateful to Mihai Codescu for the idea to relax one of the conditions of Theorem 3.1 in order to capture also the interpolation for higher order logic, and to the anonymous referees for their constructive criticism that have led to the improvement of the paper.

7. Conclusions

We have developed a general ‘borrowing’ interpolation result which transports (backwards) interpolation properties along institution comorphisms together with several methods to use this general result either within the context of an encoding or of an embedding comorphism. We have illustrated the applicability power of our method(s) by deriving interpolation results for partial algebra, preordered algebra and its Horn clause fragment, for universal sentences, and for higher order logic (with Henkin semantics), from interpolation properties of many sorted first order logic with equality and its Horn clause fragment. Based on the already existing encoding comorphisms in the literature, we strongly expect similar results in membership algebra [50] and in some of the types of order sorted algebra surveyed in [35], as well as in their corresponding Horn clause fragments. Due to lack of space we have not included the development of these result here, but we encourage the interested reader to do this by herself or himself by following the method introduced

in Section 4. The forthcoming paper [16] uses our borrowing interpolation method to derive interpolation results in the logic of predefined types [17, 23]. Moreover we expect our method to be also applicable for most of the multitude of combinations between some of the logics discussed here (such as partial preordered algebra, etc.). Most of the results developed in this paper or just mentioned now appear as new meaningful interpolation results in logics that play a significant role in computing science.

Based upon the methods of applying our general borrowing result that have been presented in this paper, we foresee the possibility to derive new interpolation results in many other logical systems. This would be an interesting topic for further work.

References

- [1] Eyal Amir and Sheila McIlraith. Improving the efficiency of reasoning through structure-based reformulation. In B.Y. Choueiry and T.Walsh, editors, *Proceedings of the Symposium on Abstraction, Reformulation and Approximation (SARA'2000)*, volume 1864 of *Lecture Notes in Artificial Intelligence*, pages 247–259. Springer-Verlag Berlin Heidelberg, 2000.
- [2] Hajnal Andréka and István Németi. A general axiomatizability theorem formulated in terms of cone-injective subcategories. In B. Csakany, E. Fried, and E.T. Schmidt, editors, *Universal Algebra*, pages 13–35. North-Holland, 1981. *Colloquia Mathematica Societas János Bolyai*, 29.
- [3] Peter B. Andrews. *An Introduction to Mathematical Logic and Type Theory*. Kluwer Academic Publishers, Dordrecht, 2nd edition, 2002.
- [4] Edigio Astesiano, Michel Bidoit, Hélène Kirchner, Berndt Krieg-Brückner, Peter Mosses, Don Sannella, and Andrzej Tarlecki. CASL: The common algebraic specification language. *Theoretical Computer Science*, 286(2):153–196, 2002.
- [5] Jan Bergstra, Jan Heering, and Paul Klint. Module algebra. *Journal of the Association for Computing Machinery*, 37(2):335–372, 1990.
- [6] Michel Bidoit and Rolf Hennicker. On the integration of the observability and reachability concepts. In *Proc. 5th Int. Conf. Foundations of Software Science and Computation Structures (FOSSACS'2002)*, volume 2303 of *Lecture Notes in Computer Science*, pages 21–36, 2002.
- [7] Tomasz Borzyszkowski. Generalized interpolation in CASL. *Information Processing Letters*, 76:19–24, 2001.
- [8] Tomasz Borzyszkowski. Logical systems for structured specifications. *Theoretical Computer Science*, 286(2):197–245, 2002.
- [9] Rod Burstall and Joseph Goguen. The semantics of Clear, a specification language. In Dines Bjorner, editor, *1979 Copenhagen Winter School on Abstract Software Specification*, volume 86 of *Lecture Notes in Computer Science*, pages 292–332. Springer, 1980.
- [10] Maura Cerioli and José Meseguer. May I borrow your logic? (transporting logical structures along maps). *Theoretical Computer Science*, 173:311–347, 1997.
- [11] Chen-Chung Chang and H. Jerome Keisler. *Model Theory*. North Holland, Amsterdam, 1990.
- [12] Alonzo Church. A formulation of the simple theory of types. *Journal of Symbolic Logic*, 5(1):56–68, 1940.
- [13] Corina Cîrstea. An institution of modal logics for coalgebras. *Logic and Algebraic Programming*, 67(1-1):87–113, 2006.
- [14] Mihai Codescu. The model theory of higher order logic. Master's thesis, Școala Normală Superioară București, 2007.
- [15] Mihai Codescu and Daniel Găină. Birkhoff completeness in institutions. *Logica Universalis*, 2(2):277–309, 2008.
- [16] Răzvan Diaconescu. Interpolation for predefined types. Submitted.
- [17] Răzvan Diaconescu. Category-based constraint logic. *Mathematical Structures in Computer Science*, 10(3):373–407, 2000.
- [18] Răzvan Diaconescu. Institution-independent ultraproducts. *Fundamenta Informaticæ*, 55(3-4):321–348, 2003.
- [19] Răzvan Diaconescu. Elementary diagrams in institutions. *Journal of Logic and Computation*, 14(5):651–674, 2004.
- [20] Răzvan Diaconescu. An institution-independent proof of Craig Interpolation Theorem. *Studia Logica*, 77(1):59–79, 2004.
- [21] Răzvan Diaconescu. Interpolation in Grothendieck institutions. *Theoretical Computer Science*, 311:439–461, 2004.
- [22] Răzvan Diaconescu. Proof systems for institutional logic. *Journal of Logic and Computation*, 16(3):339–357, 2006.
- [23] Răzvan Diaconescu. *Institution-independent Model Theory*. Birkhäuser, 2008.
- [24] Răzvan Diaconescu. An encoding of partial algebras as total algebras. *Information Processing Letters*, 109(23–24):1245–1251, 2009.
- [25] Răzvan Diaconescu and Kokichi Futatsugi. *CafeOBJ Report: The Language, Proof Techniques, and Methodologies for Object-Oriented Algebraic Specification*, volume 6 of *AMAST Series in Computing*. World Scientific, 1998.
- [26] Răzvan Diaconescu and Kokichi Futatsugi. Logical foundations of CafeOBJ. *Theoretical Computer Science*, 285:289–318, 2002.

- [27] Răzvan Diaconescu, Joseph Goguen, and Petros Stefaneas. Logical support for modularisation. In Gerard Huet and Gordon Plotkin, editors, *Logical Environments*, pages 83–130. Cambridge, 1993. Proceedings of a Workshop held in Edinburgh, Scotland, May 1991.
- [28] Răzvan Diaconescu and Petros Stefaneas. Ultraproducts and possible worlds semantics in institutions. *Theoretical Computer Science*, 379(1):210–230, 2007.
- [29] Theodosios Dimitrakos. *Formal Support for Specification Design and Implementation*. PhD thesis, Imperial College, 1998.
- [30] Theodosios Dimitrakos and Tom Maibaum. On the role of interpolation for stepwise refinement, 1997. In *Proceeding First Panhellenic Logic Symposium*.
- [31] Theodosios Dimitrakos and Tom Maibaum. On a generalized modularization theorem. *Information Processing Letters*, 74:65–71, 2000.
- [32] José L. Fiadeiro and José F. Costa. Mirror, mirror in my hand: A duality between specifications and models of process behaviour. *Mathematical Structures in Computer Science*, 6(4):353–373, 1996.
- [33] Dov M. Gabbay and Larisa Maksimova. *Interpolation and Definability: modal and intuitionistic logics*. Oxford University Press, 2005.
- [34] Joseph Goguen and Rod Burstall. Institutions: Abstract model theory for specification and programming. *Journal of the Association for Computing Machinery*, 39(1):95–146, 1992.
- [35] Joseph Goguen and Răzvan Diaconescu. An Oxford survey of order sorted algebra. *Mathematical Structures in Computer Science*, 4(4):363–392, 1994.
- [36] Joseph Goguen and Răzvan Diaconescu. Towards an algebraic semantics for the object paradigm. In Hartmut Ehrig and Fernando Orejas, editors, *Recent Trends in Data Type Specification*, volume 785 of *Lecture Notes in Computer Science*, pages 1–34. Springer, 1994.
- [37] Joseph Goguen and Grigore Roşu. Institution morphisms. *Formal Aspects of Computing*, 13:274–307, 2002.
- [38] Daniel Găină and Andrei Popescu. An institution-independent generalization of Tarski’s Elementary Chain Theorem. *Journal of Logic and Computation*, 16(6):713–735, 2006.
- [39] Daniel Găină and Andrei Popescu. An institution-independent proof of Robinson consistency theorem. *Studia Logica*, 85(1):41–73, 2007.
- [40] Leon Henkin. Completeness in the theory of types. *Journal of Symbolic Logic*, 15(2):81–91, June 1950.
- [41] Ranjit Jhala, Rupak Majumdar, and Ru-Gang Xu. State of the union: Type inference via Craig interpolation. In *Tools and Algorithms for the Construction and Analysis of Systems*, volume 4424 of *Lecture Notes in Computer Science*, pages 553–567. Springer, 2007.
- [42] Oliver Kutz and Till Mossakowski. Modules in transition. Conservativity, Composition, and Colimits. In *Proceedings, Second International Workshop on Modular Ontologies*, 2007.
- [43] Joachim Lambek and Phil Scott. *Introduction to Higher Order Categorical Logic*. Cambridge, 1986. Cambridge Studies in Advanced Mathematics, Volume 7.
- [44] Yngve Lamo. *The Institution of Multialgebras – a general framework for algebraic software development*. PhD thesis, University of Bergen, 2003.
- [45] Saunders Mac Lane. *Categories for the Working Mathematician*. Springer, second edition, 1998.
- [46] Sheila McIlraith and Eyal Amir. Theorem proving with structured theories. In *Proceedings of the 17th Intl. Conf. on Artificial Intelligence (IJCAI-01)*, pages 624 – 631, 2001.
- [47] Kenneth McMillan. Applications of Craig interpolants in model checking. In *Proceedings TACAS’2005*, volume 3440 of *Lecture Notes in Computer Science*, pages 1–12. Springer, 2005.
- [48] José Meseguer. General logics. In H.-D. Ebbinghaus et al., editors, *Proceedings, Logic Colloquium, 1987*, pages 275–329. North-Holland, 1989.
- [49] José Meseguer. Conditional rewriting logic as a unified model of concurrency. *Theoretical Computer Science*, 96(1):73–155, 1992.
- [50] José Meseguer. Membership algebra as a logical framework for equational specification. In F. Parisi-Presicce, editor, *Proc. WADT’97*, volume 1376 of *Lecture Notes in Computer Science*, pages 18–61. Springer, 1998.
- [51] Bernhard Möller, Andrzej Tarlecki, and Martin Wirsing. Algebraic specifications of reachable higher-order algebras. In Donald Sannella and Andrzej Tarlecki, editors, *Recent Trends in Data Type Specification*, volume 332 of *Lecture Notes in Computer Science*, pages 154–169. Springer Verlag, Berlin, 1987.
- [52] Till Mossakowski. Relating CASL with other specification languages: the institution level. *Theoretical Computer Science*, 286:367–475, 2002.
- [53] Till Mossakowski, Joseph Goguen, Răzvan Diaconescu, and Andrzej Tarlecki. What is a logic? In Jean-Yves Béziau, editor, *Logica Universalis*, pages 113–133. Birkhäuser, 2005.
- [54] Greg Nelson and Derek Oppen. Simplification by cooperating decision procedures. *ACM Trans. Program. Lang. Syst.*, 1(2):245–257, 1979.

- [55] Derek Oppen. Complexity, convexity and combinations of theories. *Theoretical Computer Science*, 12:291–302, 1980.
- [56] Marius Petria. An institutional version of Gödel Completeness Theorem. In *Algebra and Coalgebra in Computer Science*, volume 4624, pages 409–424. Springer Berlin / Heidelberg, 2007.
- [57] Marius Petria and Răzvan Diaconescu. Abstract Beth definability in institutions. *Journal of Symbolic Logic*, 71(3):1002–1028, 2006.
- [58] Andrei Popescu, Traian Şerbănuţă and Grigore Roşu. A semantic approach to interpolation. In *Foundations of software science and computation structures*, volume 3921 of *Lecture Notes in Computer Science*, pages 307–321. Springer Verlag, 2006.
- [59] Grigore Roşu and Joseph Goguen. On equational Craig interpolation. *Universal Computer Science*, 5(8):482–493, 1999.
- [60] Pieter-Hendrik Rodenburg. A simple algebraic proof of the equational interpolation theorem. *Algebra Universalis*, 28:48–51, 1991.
- [61] Donald Sannella and Andrzej Tarlecki. Specifications in an arbitrary institution. *Information and Control*, 76:165–210, 1988.
- [62] Lutz Schröder, Till Mossakowski, and Christoph Lüth. Type class polymorphism in an institutional framework. In José Fiadeiro, editor, *Recent Trends in Algebraic Development Techniques, 17th Intl. Workshop (WADT 2004)*, volume 3423 of *Lecture Notes in Computer Science*, pages 234–248. Springer, Berlin, 2004.
- [63] Joseph Shoenfield. *Mathematical Logic*. Addison-Wesley, 1967.
- [64] Andrzej Tarlecki. Bits and pieces of the theory of institution. In David Pitt, Samson Abramsky, Axel Poigné, and David Rydeheard, editors, *Proceedings, Summer Workshop on Category Theory and Computer Programming*, volume 240 of *Lecture Notes in Computer Science*, pages 334–360. Springer, 1986.
- [65] Andrzej Tarlecki. On the existence of free models in abstract algebraic institutions. *Theoretical Computer Science*, 37:269–304, 1986.
- [66] Andrzej Tarlecki. Quasi-varieties in abstract algebraic institutions. *Journal of Computer and System Sciences*, 33(3):333–360, 1986.
- [67] Andrzej Tarlecki. Moving between logical systems. In Magne Haveraaen, Olaf Owe, and Ole-Johan Dahl, editors, *Recent Trends in Data Type Specification*, volume 1130 of *Lecture Notes in Computer Science*, pages 478–502. Springer, 1996.
- [68] Andrzej Tarlecki. Towards heterogeneous specifications. In D. Gabbay and M. van Rijke, editors, *Proceedings, International Conference on Frontiers of Combining Systems (FroCoS'98)*, pages 337–360. Research Studies Press, 2000.
- [69] Paulo Veloso. On pushout consistency, modularity and interpolation for logical specifications. *Information Processing Letters*, 60(2):59–66, 1996.