A Semantic Approach to Interpolation

Andrei Popescu⋆ and Traian Florin Şerbănuţă⋆⋆ and Grigore Roşu

Department of Computer Science,
University of Illinois at Urbana-Champaign.
{popescu2,tserban2,grosu}@cs.uiuc.edu

Abstract. Interpolation results are investigated for various types of formulae. By shifting the focus from syntactic to *semantic interpolation*, we generate, prove and classify more than twenty interpolation results for first-order logic and some for richer logics. A few of these results non-trivially generalize known interpolation results. All the others are new.

1 Introduction

Craig interpolation is a landmark result in first-order logic [8]. In its original formulation, it says that given sentences Γ_1 and Γ_2 such that $\Gamma_1 \vdash \Gamma_2$, there is some sentence Γ whose non-logical symbols occur in *both* Γ_1 and Γ_2, called an *interpolant*, such that $\Gamma_1 \vdash \Gamma$ and $\Gamma \vdash \Gamma_2$. This well-known result can also be rephrased as follows: given first-order signatures Σ_1 and Σ_2, a Σ_1-sentence Γ_1 and a Σ_2-sentence Γ_2 such that $\Gamma_1 \models_{\Sigma_1 \cup \Sigma_2} \Gamma_2$, there is some $(\Sigma_1 \cap \Sigma_2)$-sentence Γ such that $\Gamma_1 \models_{\Sigma_1} \Gamma$ and $\Gamma \models_{\Sigma_2} \Gamma_2$.

One naturally looks for this property in logical systems others than first-order logic. The conclusion of studying various extensions of first-order logic was that "interpolation is indeed [a] rare [property in logical systems]" ([2], page 68). We are going to show in this paper that the situation is totally different when one looks in the opposite direction, at restrictions of first-order logic. There are simple logics, such as equational logic, where the interpolation result does *not* hold for sentences, but it holds for *sets of sentences* [29]. For this reason, as well as for reasons coming from theoretical software engineering, in particular from specification theory and modularization [3, 14, 16, 10], it is quite common today to state interpolation more generally, in terms of sets of sentences Γ_1, Γ_2, and Γ. This is also the approach that we follow in this paper.

We call our approach to interpolation "semantic" because we shift the problem of finding syntactic interpolants Γ to a problem of finding appropriate *classes of models*, which we call *semantic interpolants*. We present a precise characterization for *all* the semantic interpolants of a given instance $\Gamma_1 \models_{\Sigma_1 \cup \Sigma_2} \Gamma_2$, as

⋆ Also: Institute of Mathematics "Simion Stoilow" of the Romanian Academy, Bucharest; and Fundamentals of Computer Science, Faculty of Mathematics, University of Bucharest.

⋆⋆ Also: Fundamentals of Computer Science, Faculty of Mathematics, University of Bucharest.

well as a general theorem ensuring the existence of semantic interpolants closed under generic closure operators. Not all semantic interpolants correspond to sets of sentences. However, when semantic interpolants are closed under certain operators, they become *axiomatizable*, thus corresponding to some sets of sentences. Following the nice idea of using Birkhoff-like axiomatizability to prove the Craig interpolation for equational logics in [29], a similar semantic approach was investigated in [28], but it was only applied there to obtain Craig interpolation results for categorical generalizations of equational logics. A similar idea is exploited in [10], where interpolation results are presented in an institutional [19] setting. While the institution-independent interpolation results in [10] can potentially be applied to various particular logics, their instances still refer to just one type of sentence: the one the particular logic comes with.

The conceptual novelty of our semantic approach to interpolation in this paper is to keep the restrictions on Γ_1, Γ_2, and Γ, or more precisely the ones on their corresponding classes of models, *independent*. This way, surprising and interesting results can be obtained with respect to the three types of sentences involved. By considering several combinations of closure operators allowed by our parametric semantic interpolation theorem, we provide more than twenty different interpolation results; some of them generalize known results, but most of them are new. For example, we show that if the sentences in Γ_1 are first-order while the ones in Γ_2 are universally quantified Horn clauses (UHC's), then those in the interpolant Γ can be chosen to be UHC's too. Surprisingly, sometimes the interpolant is strictly simpler than Γ_1 and Γ_2. For example, we show that the following choices of the type of sentences in the interpolant Γ are possible (see also table 1):
- Γ_1-UHC's and Γ_2-positive (i.e., contains only fromulae without negations) imply that Γ consists only of universally quantified atoms;
- Γ_1-universal and Γ_2-positive imply Γ consists only of universally quantified disjunction of atoms;
- Γ_1- finitary formulae and Γ_2- infinitary UHC's imply Γ- (finitary) UHC's.

Some Motivation. Craig interpolation has applications in various areas of computer science. In automatic reasoning, *putting theories together* while still taking advantage, inside their union language, of their available decision procedures [25, 27], relies on interpolation in a crucial way. Moreover, interpolation provides a heuristic to "divide and conquer" a proving task: in order to show $\Gamma_1 \models_{\Sigma_1 \cup \Sigma_2} \Gamma_2$, find some Γ over the syntax $\Sigma_1 \cap \Sigma_2$ and prove the two "simpler" tasks $\Gamma_1 \models_{\Sigma_1} \Gamma$ and $\Gamma \models_{\Sigma_2} \Gamma_2$. For some simpler sub-logics of first-order logic, such as propositional calculus, where there is a finite set of semantically different sentences over any given signature, one can use interpolation also as a disproof technique: if for each $(\Sigma_1 \cap \Sigma_2)$-sentence Γ (there is only a finite number of them) at least one of $\Gamma_1 \models_{\Sigma_1} \Gamma$ or $\Gamma \models_{\Sigma_2} \Gamma_2$ fails, then $\Gamma_1 \models_{\Sigma_1 \cup \Sigma_2} \Gamma_2$ fails. The results of the present paper, although not effectively constructing interpolants, provide information about the existence of interpolants *of a certain type*, helping reducing the space of search. For instance, according to one of the cases of our main result, Theorem 2, the existence of a positive interpolant Γ is ensured by the fact that *either* one of Γ_1 or Γ_2 is positive.

Formal specification theory is another area where interpolation is important [22, 16] and where our results contribute. For structured specifications [4, 31], interpolation ensures a good, compositional, behavior of their semantics [4, 6, 28]. In choosing a logical framework for specifications, one has to find the right balance between expressive power and amenable computational aspects. Therefore, an intermediate choice between the "extremes", full first-order logic and equational logic, might be desirable. We enable (at least partially) such intermediate logics (e.g., the *positive*- or $(\forall\vee)$- logic) as specification frameworks, by showing that they have the interpolation property. Moreover, the very general nature of our results w.r.t. signature morphisms sometimes allows one to enrich the class of morphisms used for renaming usually up to arbitrary morphisms, freeing specifications from unnatural constraints, like injectivity of renaming/translation. Some technical details about the applications of our results to formal specifications may be found in Section 5.

Technical Preliminaries. For simplifying the exposition, set-theoretical foundational issues are ignored in this paper.[1] Given a class $\mathcal{D}$, let $\mathcal{P}(\mathcal{D})$ denote the collection of all subclasses of $\mathcal{D}$. For any $\mathcal{C} \in \mathcal{P}(\mathcal{D})$, let $\overline{\mathcal{C}}$ denote $\mathcal{D} \setminus \mathcal{C}$, that is, the class of all elements in $\mathcal{D}$ which are not in $\mathcal{C}$. Anytime we use the notation $\overline{\mathcal{C}}$, the containing class will be clear from the context. Given $\mathcal{C}_1, \mathcal{C}_2 \in \mathcal{P}(\mathcal{D})$, let $[\mathcal{C}_1, \mathcal{C}_2]$ denote the collection of all classes including $\mathcal{C}_1$ and included in $\mathcal{C}_2$, that is, $[\mathcal{C}_1, \mathcal{C}_2] = \{\mathcal{C} \in \mathcal{P}(\mathcal{D}) \mid \mathcal{C}_1 \subseteq \mathcal{C} \subseteq \mathcal{C}_2\}$.

An *operator* on $\mathcal{D}$ is a mapping $F : \mathcal{P}(\mathcal{D}) \to \mathcal{P}(\mathcal{D})$. Let $Id_{\mathcal{D}}$ denote the identity operator. For any operator F on $\mathcal{D}$, let $Fixed(F)$ denote the collection of all *fixed points* of F, that is, $\mathcal{C} \in Fixed(F)$ iff $F(\mathcal{C}) = \mathcal{C}$. An operator F on $\mathcal{D}$ is a *closure operator* iff it is *extensive* ($\mathcal{C} \subseteq F(\mathcal{C})$), *monotone* (if $\mathcal{C}_1 \subseteq \mathcal{C}_2$ then $F(\mathcal{C}_1) \subseteq F(\mathcal{C}_2)$) and *idempotent* ($F(F(\mathcal{C})) = F(\mathcal{C})$).

Given a relation $\mathcal{R}$ on $\mathcal{D}$, let $\mathcal{R}$ also denote the operator on $\mathcal{D}$ associated with $\mathcal{R}$, assigning to each $\mathcal{C} \in \mathcal{P}(\mathcal{D})$ the class of all elements in $\mathcal{D}$ which are in relation with elements in $\mathcal{C}$, that is, $\mathcal{R}(\mathcal{C}) = \{c' \in \mathcal{D} \mid (\exists c \in \mathcal{C})\ c\ \mathcal{R}\ c'\}$.

Proposition 1. *If a relation $\mathcal{R}$ on $\mathcal{D}$ is reflexive and transitive then the operator associated with $\mathcal{R}$ is a closure operator.*

Proof. Extensivity comes from reflexivity, idempotency comes from transitivity and extensivity, and monotony holds trivially, by the definition of the associated operator. $\square$

Given two classes $\mathcal{C}$ and $\mathcal{D}$ and a mapping $\mathcal{U} : \mathcal{C} \to \mathcal{D}$, we let $\mathcal{U}$ also denote the mapping $\mathcal{U} : \mathcal{P}(\mathcal{C}) \to \mathcal{P}(\mathcal{D})$ defined by $\mathcal{U}(\mathcal{C}') = \{\mathcal{U}(c) \mid c \in \mathcal{C}'\}$ for any $\mathcal{C}' \in \mathcal{P}(\mathcal{C})$. Also, we let $\mathcal{U}^{-1} : \mathcal{P}(\mathcal{D}) \to \mathcal{P}(\mathcal{C})$ denote the mapping defined by $\mathcal{U}^{-1}(\mathcal{D}') = \{c \in \mathcal{C} \mid \mathcal{U}(c) \in \mathcal{D}'\}$ for any $\mathcal{D}' \in \mathcal{P}(\mathcal{D})$. Given two mappings

[1] Yet, it is easy to see that references to collections of classes could be easily avoided.

$U, V : \mathcal{P}(\mathcal{C}) \to \mathcal{P}(\mathcal{D})$, we say that U *is included in* V, written $U \sqsubseteq V$, iff $U(\mathcal{C}') \subseteq V(\mathcal{C}')$ for any $\mathcal{C}' \in \mathcal{P}(\mathcal{C})$.

We write the composition of mappings in "diagrammatic order": if $f : A \to B$ and $g : B \to C$ then $f; g$ denotes their composition, regardless of whether f and g are mappings between sets, classes, or collections of classes.

Proposition 2. *Let $\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D}$ be classes and consider the following diagram:*

$$\begin{tikzcd} \mathcal{P}(\mathcal{A}) \arrow[r, "V"] & \mathcal{P}(\mathcal{B}) \arrow[r, bend left, "U"] \arrow[r, bend right, "U'"'] & \mathcal{P}(\mathcal{C}) \arrow[r, "V'"] & \mathcal{P}(\mathcal{D}) \end{tikzcd}$$

such that $U \sqsubseteq U'$. Then:
1. $V\,;U \sqsubseteq V\,;U'$;
2. V' monotone implies that $U\,;V' \sqsubseteq U'\,;V'$.

Proof. Both properties are obvious. $\square$

Proposition 3. *Let F and G be operators on the same class $\mathcal{D}$ such that F is a closure operator. The following hold:*
1. If $G\,;F \sqsubseteq F\,;G$ then $F\,;G\,;F = F\,;G$;
2. If G is also a closure operator, then $F\,;G$ is a closure operator iff $G\,;F \sqsubseteq F\,;G$.

Proof. 1. We use Proposition 2, together with the idempotency and extensivity of F: $F\,;(G\,;F) \sqsubseteq F\,;(F\,;G) = (F\,;F)\,;G = F\,;G \sqsubseteq (F\,;G)\,;F$.
2. Suppose that G is also a closure operator. First notice that monotony and extensivity are preserved by operator composition. Furthermore, since $G\,;F \sqsubseteq F\,;G$, we get $F\,;G\,;F\,;G \sqsubseteq F\,;F\,;G\,;G = F\,;G$, so, using also extensivity, we obtain $F\,;G$ idempotent. It follows that $F\,;G$ is a closure operator. Conversely, by extensivity of F we have $1_{\mathcal{P}(\mathcal{D})} \sqsubseteq F$. Now, since $G\,;F$ is monotone, it follows from Proposition 2 that $G\,;F \sqsubseteq F\,;G\,;F$. But since G is extensive and $F\,;G$ is idempotent, we further have $F\,;G\,;F \sqsubseteq F\,;G\,;F\,;G = F\,;G$. $\square$

Definition 1. *We say mappings (between classes) $\mathcal{U}$, $\mathcal{V}$, $\mathcal{U}'$, $\mathcal{V}'$ (see diagram) form a* ***commutative square*** *iff $\mathcal{V}'\,;\mathcal{U} = \mathcal{U}'\,;\mathcal{V}$. A commutative square is a* ***weak amalgamation square*** *iff for any $b \in \mathcal{B}$ and $c \in \mathcal{C}$ such that $\mathcal{U}(b) = \mathcal{V}(c)$, there exists some $a' \in \mathcal{A}'$ such that $\mathcal{V}'(a') = b$ and $\mathcal{U}'(a') = c$.*

$$\begin{tikzcd} & \mathcal{A} & \\ \mathcal{B} \arrow[ur, "\mathcal{U}"] & & \mathcal{C} \arrow[ul, "\mathcal{V}"'] \\ & \mathcal{A}' \arrow[ul, "\mathcal{V}'"] \arrow[ur, "\mathcal{U}'"'] & \end{tikzcd}$$

We call this amalgamation square "weak" because a' is not required to be unique.

2 First-Order Logic and Classical Interpolation Revisited

First-Order Logic. A first-order signature is a triple $\Sigma = (S, F, P)$, where S is a set of *sorts*, $(F_{w\to s})_{w\in S^*, s\in S}$ an $S^* \times S$-ranked set of *operation symbols*, and $P = (P_w)_{w\in S^*}$ an S^*-ranked set of *relation symbols*. By language abuse, we also write P and F for the sets $\bigcup_{w\in S^*} P_w$ and $\bigcup_{w\in S^*,s\in S} F_{w\to s}$ respectively. The Σ-*models* are structures consisting of a non-empty carrier set for each sort,[2] and of an operation/relation of appropriate arity for each symbol of operation/relation. Formally, models are triples

$$A = ((A_s)_{s\in S}, (A_{w\to s}(\sigma))_{(w,s)\in S^*\times S,\sigma\in F_{w\to s}}, (A_w(\pi))_{w\in S^*,\pi\in P_w}),$$

where $A_{w\to s}(\sigma) : A^w \to A_s$ if $\sigma \in F_{w\to s}$ and $A_w(\pi) \subseteq A^w$ if $\pi \in P_w$. (A^w denotes $A_{s_1} \times \ldots \times A_{s_n}$ whenever $w = s_1 \ldots s_n \in S^*$.) We sometimes write A_σ and A_π instead of $A_{w\to s}(\sigma)$ and $A_w(\pi)$. We let $Mod(\Sigma)$ denote the class of Σ-models. The set of (first-order) Σ-sentences is obtained starting with atomic formulae, that is, relational atoms $\pi(t_1, \ldots, t_n)$ and equational atoms $t_1 = t_2$ ($t_1, \ldots, t_n$ are terms over some component-wise infinite S-sorted set of variables and operations in F), and using negation, conjunction, disjunction, universal and existential quantification, such that each variable gets to be bounded by some quantifier. Let $Sen(\Sigma)$ denote the set of Σ-sentences. The *satisfaction relation* $A \models \gamma$ between models and sentences is defined as usual, by interpreting the syntactic items in γ as the corresponding items in A. The satisfaction relation can be extended to a relation $\models$ between classes of models $\mathcal{M} \subseteq Mod(\Sigma)$ and sets of sentences $\Gamma \subseteq Sen(\Sigma)$: $\mathcal{M} \models \Gamma$ iff $A \models \gamma$ for all $A \in \mathcal{M}$ and $\gamma \in \Gamma$. This further induces two operators $\_^* : \mathcal{P}(Sen(\Sigma)) \to \mathcal{P}(Mod(\Sigma))$ and $\_^* : \mathcal{P}(Mod(\Sigma)) \to \mathcal{P}(Sen(\Sigma))$, defined by $\Gamma^* = \{A \mid \{A\} \models \Gamma\}$ and $\mathcal{M}^* = \{\gamma \mid \mathcal{M} \models \{\gamma\}\}$ for each $\Gamma \subseteq Sen(\Sigma)$ and $\mathcal{M} \subseteq Mod(\Sigma)$. The two operators $\_^*$ form a Galois connection between $(\mathcal{P}(Sen(\Sigma)), \subseteq)$ and $(\mathcal{P}(Mod(\Sigma)), \subseteq)$. The two composition operators $\_^* ; \_^*$ are denoted $\_^\bullet$ and are called *deduction closure* (the one on sets of sentences) and *axiomatizable hull* (the one on classes of models). The classes of models closed under $\_^\bullet$ are called *elementary classes* and the sets of sentences closed under $\_^\bullet$ are called *theories*. If $\Gamma, \Gamma' \subseteq Sen(\Sigma)$, we say that Γ *semantically deduces* Γ', written $\Gamma \models \Gamma'$, iff $\Gamma^* \subseteq \Gamma'^*$.

Signature morphisms are mappings on sorts, together with rank-preserving mappings on operation and relation symbols. More precisely, given two signatures $\Sigma = (S, F, P)$ and $\Sigma' = (S', F', P')$, a signature morphism $\phi : \Sigma \to \Sigma'$ is a triple $(\phi^{st}, \phi^{op}, \phi^{rl})$, where $\phi^{st} : S \to S'$, $\phi^{op} = (\phi^{op}_{w\to s})_{w\in S^*,s\in S}$, $\phi^{rl} = (\phi^{rl}_w)_{w\in S^*}$ such that for each w and s, $\phi^{op}_{w\to s} : F_{w\to s} \to F'_{\phi^{st}(w)\to\phi^{st}(s)}$ and $\phi^{rl}_w : P_w \to P'_{\phi^{st}(w)}$. We may write ϕ for each one of ϕ^{st}, ϕ^{op} and ϕ^{rl}. The morphism ϕ naturally induces a mapping $\phi : Sen(\Sigma) \to Sen(\Sigma')$, where for each Σ-sentence γ, $\phi(\gamma)$ is obtained from γ by appropriately renaming via ϕ all the sorts, operation- and relation- symbols. For each Σ'-model A', its *ϕ-reduct* is a model A, usually written as $A' \restriction_\phi$, where $A_s = A'_{\phi(s)}$, $A_{w\to s}(\sigma) = A'_{\phi(w)\to\phi(s)}(\phi(\sigma))$, and $A_w(\pi) = A'_{\phi(w)}(\phi(\pi))$. Let $Mod(\phi) : Mod(\Sigma') \to Mod(\Sigma)$ denote the mapping $A' \mapsto A'\restriction_\phi$. Satisfaction relation has the important property that it is "invariant under change of notation" [19], i.e., for each $\gamma \in Sen(\Sigma)$ and $A' \in Mod(\Sigma')$, $A' \models \phi(\gamma)$ iff $A'\restriction_\phi \models \gamma$.

[2] Birkhoff-style axiomatizability, which will be used intensively in this paper, depends on the non-emptiness of carriers [29].

Interpolation. The original formulation of interpolation [8] is in terms of signature intersections and unions, that is, w.r.t. squares which are pushouts of signature inclusions. However, subsequent advances in modularization theory [3, 14, 16, 10, 5] showed the need of arbitrary pushout squares or even weak amalgamation squares. A general formulation of interpolation is the following:

Definition 2. *Assume a commutative square of signature morphisms (see diagram) and two sets of sentences $\Gamma_1 \subseteq Sen(\Sigma_1)$, $\Gamma_2 \subseteq Sen(\Sigma_2)$ such that $\phi'_2(\Gamma_1) \models_{\Sigma'} \phi'_1(\Gamma_2)$ (i.e., Γ_1 implies Γ_2 on the "union language" Σ'). An* ***interpolant*** *for Γ_1 and Γ_2 is a set $\Gamma \subseteq Sen(\Sigma)$ such that $\Gamma_1 \models_{\Sigma_1} \phi_1(\Gamma)$ and $\phi_2(\Gamma) \models_{\Sigma_2} \Gamma_2$.*

$$\begin{tikzcd} & \Sigma \arrow[dl, "\phi_1"'] \arrow[dr, "\phi_2"] & \\ \Sigma_1 \arrow[dr, "\phi'_2"'] & & \Sigma_2 \arrow[dl, "\phi'_1"] \\ & \Sigma' & \end{tikzcd}$$

The following three examples show that, without further restrictions on signature morphisms, an interpolant Γ may not be found with **the same type** of sentences as Γ_1 and Γ_2, but with more general ones. In other words, there are sub-first-order logics which do not admit Craig Interpolation within themselves but in a larger (sub-)logic. The first example below shows a square in unconditional equational logic which does not admit unconditional interpolants, but admits a conditional one:

Example 1. Consider the following pushout of algebraic signatures, as in [28]: $\Sigma = (\{s\}, \{d_1, d_2 : s \to s\})$, $\Sigma_1 = (\{s\}, \{d_1, d_2, c : s \to s\})$, $\Sigma_2 = (\{s\}, \{d : s \to s\})$, $\Sigma' = (\{s\}, \{d, c : s \to s\})$, all morphisms mapping the sort s into itself, ϕ_1 and ϕ_2 mapping d_1 and d_2 into themselves and into d, respectively, ϕ'_2 mapping d_1 and d_2 into d and c into itself, and ϕ'_1 mapping d into itself.

Take $\Gamma_1 = \{(\forall x)d_2(x) = c(d_1(x)),\ (\forall x)d_1(d_2(x)) = c(d_2(x))\}$ and $\Gamma_2 = \{(\forall x)d(d(x)) = d(x)\}$ to be sets of Σ_1-equations and of Σ_2-equations, respectively. It is easy to see that Γ_1 implies Γ_2 in the "union language", i.e., $\phi'_2(\Gamma_1) \models \phi'_1(\Gamma_2)$. But Γ_1 and Γ_2 have no equational Σ-interpolant, because the only equational Σ-consequences of Γ_1 are the trivial ones, of the form $(\forall X)t = t$ with t a Σ-term (since all the nontrivial Σ_1-consequences of Γ_1 contain the symbol c). Yet, Γ_1 and Γ_2 have a conditional-equational interpolant, namely $\{(\forall x)d_1(x) = d_2(x) \Rightarrow d_1(x) = d_1(d_1(x))\}$.

The following example shows a situation in which the interpolant cannot even be conditional-equational; it can be a first-order, though:

Example 2. Consider the same pushout of signatures as in previous example and take now $\Gamma_1 = \{(\forall x)d_2(x) = d_1(c(x)),\ (\forall x)d_1(d_2(x)) = d_2(c(x))\}$ and $\Gamma_2 = \{(\forall x)d(d(x)) = d(x)\}$. Again, $\phi'_2(\Gamma_1) \models \phi'_1(\Gamma_2)$. But now Γ_1 and Γ_2 have no conditional-equational Σ-interpolant either, because all nontrivial conditional equations we can infer from Γ_1 contain the symbol c (to see this, think in terms of the syntactic deduction system for conditional equational logic). Nevertheless, Γ_1 and Γ_2 have a first-order interpolant, namely $\{(\forall x)d_1(x) = d_2(x) \Rightarrow (\forall y)d_1(y) = d_1(d_1(y))\}$.

An obstacle to interpolation inside the desired type of sentences in the examples above is the lack of injectivity of ϕ_2 on operation symbols; injectivity on both sorts and operation symbols implies conditional equational interpolation [29]. The following example, taken over form [5] shows that first-order logic does not admit interpolation either without making additional requirements on the square morphisms. We shall shortly prove that for a pushout square to have first-order interpolation, it is sufficient that it has *one* of the morphisms injective *on sorts*. This is, up to our knowledge, the most general known effective criterion for a pushout to have first-order interpolation (see also [21] for the same result obtained by taking a totally different route, via Robinson consistency).

Example 3. Let $\Sigma = (\{s_1, s_2\}, \{d_1 :\to s_1, d_2 :\to s_2\})$, $\Sigma_1 = (\{s\}, \{d_1, d_2 :\to s\})$, $\Sigma_2 = (\{s\}, \{d :\to s\})$, $\Sigma' = (\{s\}, \{d :\to s\})$, all the morphisms mapping all sorts into s, ϕ_1 mapping d_1 and d_2 into themselves, and all the other morphisms mapping all operation symbols into d. In [5], it is shown that first-order interpolation does not hold in this context. For instance, let $\Gamma_1 = \{\neg(d_1 = d_2)\}$ and $\Gamma_2 = \{\neg(d = d)\}$. Then obviously $\phi_2'(\Gamma_1) \models \phi_1'(\Gamma_2)$, but Γ_1 and Γ_2 have no (first-order) Σ-interpolant. Indeed, assume by contradiction that there exists a set Γ of Σ-sentences such that $\Gamma_1 \models \phi_1(\Gamma)$ and $\phi_2(\Gamma) \models \Gamma_2$; let A be the Σ_1-model with $A_s = \{0, 1\}$, such that $A_{d_1} = 0$ and $A_{d_2} = 1$. Let B denote $A\lceil_{\phi_1}$. We have that $B_{s_1} = B_{s_2} = \{0, 1\}$, $B_{d_1} = 0$, $B_{d_2} = 1$. Because $A \models \Gamma_1$ and $\Gamma_1 \models \phi_1(\Gamma)$, it holds that $B \models \Gamma$. Define the Σ-model C to be the same as B, just that one takes $C_{d_1} = C_{d_2} = 0$. Now, C and B are isomorphic (notice that a and b are constants of different sorts in Σ), so $C \models \Gamma$; but C admits a ϕ_2-extension D, and, because $\phi_2(\Gamma) \models \Gamma_2$, we get $D \models \Gamma_2$, which is a contradiction, since no Σ_2-model can satisfy $\neg(d = d)$. What one would need here in order to "fix" interpolation is some extension of many-sorted first-order formulae which would allow one *to equate terms of different sorts*, in the form $t_1.s_1 = t_2.s_2$; alternatively, an order-sorted second-order extension, allowing quantification over sorts, a special symbol $<$ which is to be interpreted as inclusion between sort carriers, and membership assertions $t : s$, meaning "t is of sort s" (in the spirit of [23]), would do, because we could formally state in Σ that there exists a common subsort s' of s_1 and s_2 such that $d_1 : s'$ and $\neg(d_2 : s')$.

Therefore, the interpolation property seems to be very sensitive to the limits of expression of the given logic. Analyzing interpolation w.r.t. these limits in diverse sub-first-order logics is one of the main concerns of this paper.

The semantic view to interpolation. The interpolation problem, despite its syntactic nature, can be regarded *semantically*, on classes of models. Indeed, by the sentence-model duality and the satisfaction condition, we have that:
- $\phi_2'(\Gamma_1) \models \phi_1'(\Gamma_2)$ iff $\phi_2'(\Gamma_1)^* \subseteq \phi_1'(\Gamma_2)^*$ iff $Mod(\phi_2')^{-1}(\Gamma_1^*) \subseteq Mod(\phi_1')^{-1}(\Gamma_2^*)$;
- $\Gamma_1 \models \phi_1(\Gamma)$ iff $\Gamma_1^* \subseteq \phi_1(\Gamma)^*$ iff $\Gamma_1^* \subseteq Mod(\phi_1)^{-1}(\Gamma^*)$;
- $\phi_2(\Gamma) \models \Gamma_2$ iff $\phi_2(\Gamma)^* \subseteq \Gamma_2^*$ iff $Mod(\phi_2)^{-1}(\Gamma^*) \subseteq \Gamma_2^*$.
These suggest defining the following notion of "semantic interpolation":

Definition 3. *Given $\mathcal{M}_1 \subseteq Mod(\Sigma_1)$ and $\mathcal{M}_2 \subseteq Mod(\Sigma_2)$ with $Mod(\phi_2')^{-1}(\mathcal{M}_1) \subseteq Mod(\phi_1')^{-1}(\mathcal{M}_2)$, a **semantic interpolant** of $\mathcal{M}_1$ and $\mathcal{M}_2$ is a class $\mathcal{M} \subseteq Mod(\Sigma)$ such that $\mathcal{M}_1 \subseteq Mod(\phi_1)^{-1}(\mathcal{M})$ and $Mod(\phi_2)^{-1}(\mathcal{M}) \subseteq \mathcal{M}_2$.*

Therefore, if $\Gamma_1 \subseteq Sen(\Sigma_1)$, $\Gamma_2 \subseteq Sen(\Sigma_2)$, and $\Gamma \subseteq Sen(\Sigma)$, then Γ is an interpolant of Γ_1 and Γ_2 iff Γ^* is a semantic interpolant of Γ_1^* and Γ_2^*. The connection between semantic interpolation and classical logical interpolation holds only when one considers classes which are *elementary*, i.e., specified by sets of sentences, and the interpolant is also elementary. Rephrasing the interpolation problem semantically allows us to adopt the following "divide and conquer" approach, already sketched in [28]:

1. Find as many semantic interpolants as possible without caring whether they are axiomatizable or not (note that "axiomatizable" will mean "elementary" only within first-order logic, but we shall consider other logics as well);
2. Then, by imposing diverse axiomatizability closures on the two starting classes of models, try to obtain a closed interpolant.

The two starting classes need not fulfill the same type of axiomatizability. And in the most fortunate cases, as we shall see below, the interpolant is able to capture and even strengthen the properties of both classes. Though quite non-constructive because of the appeal to classes of models, our results will give plenty of information about the interpolant. The next two sections deal with the two parts of the just divided interpolation problem.

3 Semantic Interpolation

In this section we generalize the notion of semantic interpolant from Definition 3 to arbitrary classes and give a characterization of semantic interpolants together with a theorem stating generic conditions under which such semantic interpolants are closed under certain closure operators.

Definition 4. *Consider a commutative square like in Definition 1, together with some $\mathcal{M} \in \mathcal{P}(\mathcal{B})$ and $\mathcal{N} \in \mathcal{P}(\mathcal{C})$ such that ${\mathcal{V}'}^{-1}(\mathcal{M}) \subseteq {\mathcal{U}'}^{-1}(\mathcal{N})$. We say that $\mathcal{K} \in \mathcal{P}(\mathcal{A})$ is a **semantic interpolant** of $\mathcal{M}$ and $\mathcal{N}$ iff $\mathcal{M} \subseteq \mathcal{U}^{-1}(\mathcal{K})$ and $\mathcal{V}^{-1}(\mathcal{K}) \subseteq \mathcal{N}$. Let $\mathcal{I}(\mathcal{M},\mathcal{N})$ denote the collection of all semantic interpolants of $\mathcal{M}$ and $\mathcal{N}$.*

The following gives a precise characterization of semantic interpolants together with a general condition under which they exist.

Proposition 4. *Under the hypothesis of Definition 4:*
1. $\mathcal{I}(\mathcal{M},\mathcal{N}) = [\mathcal{U}(\mathcal{M}), \overline{\mathcal{V}(\overline{\mathcal{N}})}]$;
2. If the square is a weak amalgamation square then $\mathcal{I}(\mathcal{M},\mathcal{N}) \neq \emptyset$.

Proof. 1: For any $\mathcal{K} \in \mathcal{A}$, we have that $\mathcal{M} \subseteq \mathcal{U}^{-1}(\mathcal{K})$ is equivalent to $\mathcal{U}(\mathcal{M}) \subseteq \mathcal{K}$; moreover, one can see that $\mathcal{V}^{-1}(\mathcal{K}) \subseteq \mathcal{N}$ is equivalent to $\mathcal{K} \subseteq \overline{\mathcal{V}(\overline{\mathcal{N}})}$. Therefore, $\mathcal{K}$ is a semantic interpolant for M and $\mathcal{N}$ iff $\mathcal{U}(\mathcal{M}) \subseteq \mathcal{K} \subseteq \overline{\mathcal{V}(\overline{\mathcal{N}})}$.
2: All we need to show is that $\mathcal{U}(\mathcal{M}) \subseteq \overline{\mathcal{V}(\overline{\mathcal{N}})}$, i.e., that for any $a \in \mathcal{U}(\mathcal{M})$, a is not an element of $\mathcal{V}(\overline{\mathcal{N}})$. Suppose it were and consider $b \in \mathcal{M}$ and $c \in \overline{\mathcal{N}}$ such

that $\mathcal{U}(b) = a = \mathcal{V}(c)$. From the weak amalgamation property we deduce that there exists some $a' \in \mathcal{A}'$ such that $\mathcal{V}'(a') = b$ and $\mathcal{U}'(a') = c$. Since $b \in \mathcal{M}$, it follows that $a' \in {\mathcal{V}'}^{-1}(\mathcal{M})$; since ${\mathcal{V}'}^{-1}(\mathcal{M}) \subseteq {\mathcal{U}'}^{-1}(\mathcal{N})$, it further follows that $a' \in {\mathcal{U}'}^{-1}(\mathcal{N})$, i.e., that $\mathcal{U}'(a') \in \mathcal{N}$. However, this is in contradiction with the fact that $c = \mathcal{U}'(a')$ was chosen from $\overline{\mathcal{N}}$. $\square$

In the remainder of this section, when we use pairs of operators/relations, we will annotate each component of the pair by the class on which it is defined.

Definition 5. *Given two classes $\mathcal{C}$ and $\mathcal{D}$, a mapping $\mathcal{U} : \mathcal{C} \to \mathcal{D}$ and a pair of operators $F = (F_\mathcal{C}, F_\mathcal{D})$, we say that*

- *$\mathcal{U}$ **preserves fixed points of** F if $\mathcal{U}(Fixed(F_\mathcal{C})) \subseteq Fixed(F_\mathcal{D})$, that is, for any fixed point of $F_\mathcal{C}$ we obtain through $\mathcal{U}$ a fixed point of $F_\mathcal{D}$;*
- *$\mathcal{U}$ **lifts** F if $F_\mathcal{D} \,; \mathcal{U}^{-1} \sqsubseteq \mathcal{U}^{-1} \,; F_\mathcal{C}$, that is, for any $\mathcal{D}' \in \mathcal{P}(\mathcal{D})$ and any $c \in \mathcal{C}$, if $\mathcal{U}(c) \in F_\mathcal{D}(\mathcal{D}')$ then $c \in F_\mathcal{C}(\mathcal{U}^{-1}(\mathcal{D}'))$.*

The intuition for the word "lifts" in the above definition comes from the case of the operators $F_\mathcal{C}$ and $F_\mathcal{D}$ being given by binary relations (see Proposition 5 below).

The following theorem is at the heart of all our subsequent results. It gives general criteria under which a weak amalgamation square admits semantic interpolants *closed* under certain generic operators.

Theorem 1. *Consider a weak amalgamation square as in Definition 1 and pairs of operators $F = (F_\mathcal{B}, F_\mathcal{A})$ and $G = (G_\mathcal{C}, G_\mathcal{A})$ such that:*

1. *$F_\mathcal{A} \,; G_\mathcal{A} \,; F_\mathcal{A} = F_\mathcal{A} \,; G_\mathcal{A}$;*
2. *$G_\mathcal{C}$ and $G_\mathcal{A}$ are closure operators;*
3. *$\mathcal{U}$ preserves fixed points of F;*
4. *$\mathcal{V}$ lifts G.*

Then for each $\mathcal{M} \in Fixed(F_\mathcal{B})$ and $\mathcal{N} \in Fixed(G_\mathcal{C})$ such that ${\mathcal{V}'}^{-1}(\mathcal{M}) \subseteq {\mathcal{U}'}^{-1}(\mathcal{N})$, $\mathcal{M}$ and $\mathcal{N}$ have a semantic interpolant $\mathcal{K}$ in $Fixed(F_\mathcal{A}) \cap Fixed(G_\mathcal{A})$.

Proof. Take $\mathcal{K} = G_\mathcal{A}(\mathcal{U}(\mathcal{M}))$. Let us first show that $\mathcal{K} \in Fixed(F_\mathcal{A}) \cap Fixed(G_\mathcal{A})$. We have that $\mathcal{K} \in Fixed(G_\mathcal{A})$, since $G_\mathcal{A}$ is idempotent. Also, since $\mathcal{M} \in Fixed(F_\mathcal{B})$ and $\mathcal{U}$ preserves fixed points of F, we have that $\mathcal{U}(\mathcal{M}) \in Fixed(F_\mathcal{A})$. Therefore $F_\mathcal{A}(\mathcal{K}) = F_\mathcal{A}(G_\mathcal{A}(\mathcal{U}(\mathcal{M}))) = F_\mathcal{A}(G_\mathcal{A}(F_\mathcal{A}(\mathcal{U}(\mathcal{M})))) = G_\mathcal{A}(F_\mathcal{A}(\mathcal{U}(\mathcal{M}))) = G_\mathcal{A}(\mathcal{U}(\mathcal{M})) = \mathcal{K}$, that is, $\mathcal{K} \in Fixed(F_\mathcal{A})$.

Let us next show that $\mathcal{K}$ is a semantic interpolant of $\mathcal{M}$ and $\mathcal{N}$. Since $G_\mathcal{A}$ is extensive, $\mathcal{U}(\mathcal{M}) \subseteq G_\mathcal{A}(\mathcal{U}(\mathcal{M}))$, whence $\mathcal{M} \subseteq \mathcal{U}^{-1}(\mathcal{K})$. Using that $\mathcal{V}$ lifts G, we obtain that $\mathcal{V}^{-1}(\mathcal{K}) = \mathcal{V}^{-1}(G_\mathcal{A}(\mathcal{U}(\mathcal{M}))) \subseteq G_\mathcal{C}(\mathcal{V}^{-1}(\mathcal{U}(\mathcal{M})))$. From Proposition 4 we know that $\mathcal{U}(\mathcal{M})$ is a semantic interpolant of $\mathcal{M}$ and $\mathcal{N}$, so $\mathcal{V}^{-1}(\mathcal{U}(\mathcal{M})) \subseteq \mathcal{N}$. Using that $G_\mathcal{C}$ is monotone, we get that $G_\mathcal{C}(\mathcal{V}^{-1}(\mathcal{U}(\mathcal{M}))) \subseteq G_\mathcal{C}(\mathcal{N}) = \mathcal{N}$. Thus $\mathcal{V}^{-1}(\mathcal{K}) \subseteq \mathcal{N}$. We obtained that $\mathcal{K}$ is also a semantic interpolant of $\mathcal{M}$ and $\mathcal{N}$. $\square$

The operators above will be conveniently chosen in the next section to be closure operators characterizing axiomatizable classes of models.

Because in the following sections many of the used operators are associated to (reflexive and transitive) relations, let us next give an easy criterion for a mapping to lift/[preserve fixed points of] such an operator.

Proposition 5. *Consider two classes $\mathcal{C}$ and $\mathcal{D}$, a mapping $\mathcal{U} : \mathcal{C} \to \mathcal{D}$ and a pair of relations $R = (R_\mathcal{C}, R_\mathcal{D})$.*[3] *Then the following hold:*
1. $\mathcal{U}$ lifts R if and only if for any elements $c \in \mathcal{C}$ and $d \in \mathcal{D}$ such that $dR_\mathcal{D}\mathcal{U}(c)$, there exists $c' \in \mathcal{C}$ such that $\mathcal{U}(c') = d$ and $c'R_\mathcal{C}c$.
2. Suppose $R_\mathcal{C}$ is reflexive and transitive. Then $\mathcal{U}$ preserves fixed points of R if and only if for any elements $c \in \mathcal{C}$ and $d \in \mathcal{D}$ such that $\mathcal{U}(c)R_\mathcal{D}d$, there exists $c' \in \mathcal{C}$ such that $\mathcal{U}(c') = d$ and $cR_\mathcal{C}c'$;
3. Suppose $R_\mathcal{C}$ is reflexive and transitive. Then $\mathcal{U}$ preserves fixed points of R if and only if $\mathcal{U}$ lifts $(R_\mathcal{C}^{-1}, R_\mathcal{D}^{-1})$.

Proof. 1: Assume that $\mathcal{U}$ lifts R and let $c \in \mathcal{C}$ and $d \in \mathcal{D}$ be two elements such that $dR_\mathcal{D}\mathcal{U}(c)$. Then $\mathcal{U}(c) \in R_\mathcal{D}(\{d\})$, thus $c \in R_\mathcal{C}(\mathcal{U}^{-1}(\{d\}))$, i.e., there exists $c' \in \mathcal{C}$ such that $\mathcal{U}(c') \in \{d\}$ and $c \in R_\mathcal{C}(\{c'\})$. But the latter just mean $\mathcal{U}(c') = d$ and $c'R_\mathcal{C}c$. Conversely, let $\mathcal{D}' \in \mathcal{P}(\mathcal{D})$ and $c \in \mathcal{C}$ such that $\mathcal{U}(c) \in R_\mathcal{D}(\mathcal{D}')$. Then there exists $d \in \mathcal{D}'$ such that $dR_\mathcal{D}\mathcal{U}(c)$. Thus, there exists $c' \in \mathcal{C}$ such that $\mathcal{U}(c') = d$ and $c'R_\mathcal{C}c$. But this implies $c' \in \mathcal{U}^{-1}(\mathcal{D}')$, and furthermore $c \in R_\mathcal{C}(\mathcal{U}^{-1}(\mathcal{D}'))$.

2: Suppose $\mathcal{U}$ preserves fixed points of R and let $c \in \mathcal{C}$ and $d \in \mathcal{D}$ be two elements such that $\mathcal{U}(c)R_\mathcal{D}d$. Since $R_\mathcal{C}$ is reflexive and transitive we have that $R_\mathcal{C}(\{c\})$ is a fixed point of $R_\mathcal{C}$, so $\mathcal{U}(R_\mathcal{C}(\{c\}))$ must be a fixed point of $R_\mathcal{D}$. Since $\mathcal{U}(c) \in \mathcal{U}(R_\mathcal{C}(\{c\}))$ and $\mathcal{U}(c)R_\mathcal{D}d$, it follows that $d \in \mathcal{U}(R_\mathcal{C}(\{c\}))$, whence there exist $c' \in R_\mathcal{C}(\{c\})$ such that $\mathcal{U}(c') = d$. But $c' \in R_\mathcal{C}(\{c\})$ means exactly that $cR_\mathcal{C}c'$. Conversely, let $\mathcal{C}'$ be a fixed point of $R_\mathcal{C}$. We want to show that $\mathcal{U}(\mathcal{C}')$ is a fixed point of $R_\mathcal{D}$. Let $d \in R_\mathcal{D}(\mathcal{U}(\mathcal{C}'))$. There exists $c \in \mathcal{C}'$ such that $\mathcal{U}(c)R_\mathcal{D}d$ whence there exists $c' \in \mathcal{C}$ such that $\mathcal{U}(c') = d$ and $cR_\mathcal{C}c'$. Since $c \in \mathcal{C}'$ and $\mathcal{C}'$ is a fixed point of $R_\mathcal{C}$, it follows that $c' \in \mathcal{C}'$, whence $d \in \mathcal{U}(\mathcal{C}')$.

3: Obvious from points 1 and 2. □

Since our closure operators will be in general compositions of other closure operators, the following result allows us to treat them separately.

Proposition 6. *Consider two classes $\mathcal{C}$ and $\mathcal{D}$, a mapping $\mathcal{U} : \mathcal{C} \to \mathcal{D}$ and two pairs of operators $F = (F_\mathcal{C}, F_\mathcal{D})$ and $G = (G_\mathcal{C}, G_\mathcal{D})$. Then the following hold:*
1. If $G_\mathcal{C}$ is monotone and $\mathcal{U}$ lifts F and G, then $\mathcal{U}$ also lifts $(F_\mathcal{C}\,; G_\mathcal{C}, F_\mathcal{D}\,; G_\mathcal{D})$;
2. If $F_\mathcal{C}\,; G_\mathcal{C}\,; F_\mathcal{C} = F_\mathcal{C}\,; G_\mathcal{C}$ and $\mathcal{U}$ preserves fixed points of F and G, then $\mathcal{U}$ also preserves fixed points of $(F_\mathcal{C}\,; G_\mathcal{C}, F_\mathcal{D}\,; G_\mathcal{D})$.
3. If $F_\mathcal{C}$ and $G_\mathcal{C}$ are closure operators and $F_\mathcal{C}; G_\mathcal{C}$ is idempotent, then $F_\mathcal{C}; G_\mathcal{C}; F_\mathcal{C} = F_\mathcal{C}; G_\mathcal{C}$.

[3] Recall that $R_\mathcal{C}$ and $R_\mathcal{D}$ also denote the induced operators.

Proof. 1: We use Proposition 2. First take V to be $F_{\mathcal{D}}$ and U, U' to be $G_{\mathcal{D}}\,;\mathcal{U}^{-1}$ and $\mathcal{U}^{-1}\,;G_{\mathcal{C}}$ respectively, to obtain $F_{\mathcal{D}}\,;G_{\mathcal{D}}\,;\mathcal{U}^{-1} \sqsubseteq F_{\mathcal{D}}\,;\mathcal{U}^{-1}\,;G_{\mathcal{C}}$. Next take V' to be $G_{\mathcal{C}}$ (which is monotone) and U, U' to be $F_{\mathcal{D}}\,;\mathcal{U}^{-1}$ and $\mathcal{U}^{-1}\,;F_{\mathcal{C}}$ respectively, to obtain $F_{\mathcal{D}}\,;\mathcal{U}^{-1}\,;G_{\mathcal{C}} \sqsubseteq \mathcal{U}^{-1}\,;F_{\mathcal{C}}\,;G_{\mathcal{C}}$. Thus $F_{\mathcal{D}}\,;G_{\mathcal{D}}\,;\mathcal{U}^{-1} \sqsubseteq \mathcal{U}^{-1}\,;F_{\mathcal{C}}\,;G_{\mathcal{C}}$.
2: Let $\mathcal{C}'$ be a fixed point of $F_{\mathcal{C}}\,;G_{\mathcal{C}}$, i.e., such that $G_{\mathcal{C}}(F_{\mathcal{C}}(\mathcal{C}')) = \mathcal{C}'$. ¿From $F_{\mathcal{C}}\,;G_{\mathcal{C}}\,;F_{\mathcal{C}} = F_{\mathcal{C}}\,;G_{\mathcal{C}}$, we get $F_{\mathcal{C}}(\mathcal{C}') = F_{\mathcal{C}}(G_{\mathcal{C}}(F_{\mathcal{C}}(\mathcal{C}'))) = G_{\mathcal{C}}(F_{\mathcal{C}}(\mathcal{C}')) = \mathcal{C}'$. Therefore, it is also the case that $G_{\mathcal{C}}(\mathcal{C}') = G_{\mathcal{C}}(F_{\mathcal{C}}(\mathcal{C}')) = \mathcal{C}'$. Hence, $\mathcal{C}'$ is a fixed point of both $F_{\mathcal{C}}$ and $G_{\mathcal{C}}$. Since $\mathcal{U}$ preserves fixed points of F and G, it follows that $\mathcal{U}(\mathcal{C}')$ is a fixed point of both $F_{\mathcal{D}}$ and $G_{\mathcal{D}}$, therefore $G_{\mathcal{D}}(F_{\mathcal{D}}(\mathcal{U}(\mathcal{C}'))) = G_{\mathcal{D}}(\mathcal{U}(\mathcal{C}')) = \mathcal{U}(\mathcal{C}')$, that is, $\mathcal{U}(\mathcal{C}')$ is a fixed point of $F_{\mathcal{D}}; G_{\mathcal{D}}$.
3: $F_{\mathcal{C}}; G_{\mathcal{C}} \sqsubseteq F_{\mathcal{C}}; G_{\mathcal{C}}; F_{\mathcal{C}}$ follows from $F_{\mathcal{C}}$ being extensive and $F_{\mathcal{C}}; G_{\mathcal{C}}$ monotone. On the other hand, since $G_{\mathcal{C}}$ is extensive and $F_{\mathcal{C}}; G_{\mathcal{C}}$ is idempotent, we get $F_{\mathcal{C}}; G_{\mathcal{C}}; F_{\mathcal{C}} \sqsubseteq F_{\mathcal{C}}; G_{\mathcal{C}}; F_{\mathcal{C}}; G_{\mathcal{C}} = F_{\mathcal{C}}; G_{\mathcal{C}}$. ☐

4 New Interpolation Results

In this section, we shall give a series of interpolation results for various types of first-order sentences. As mentioned, our semantic approach allows us to exploit axiomatizability results; since these results use model (homo)morphisms, we first briefly recall some definitions related to this notion. Given two Σ-models A and B, a morphism $h : A \to B$ is an S-sorted function $h = (h_s : A_s \to B_s)_{s \in S}$ that commutes with operations and preserves relations: for each $w = s_1 \ldots s_n \in S^*$, $s \in S$, $\sigma \in F_{w \to s}$, $\pi \in P_w$, and $(a_1, \ldots, a_n) \in A_w$, it is the case that $h_s(A_\sigma(a_1, \ldots a_n)) = B_\sigma(h_{s_1}(a_1), \ldots h_{s_n}(a_n))$ and $[(a_1, \ldots, a_n) \in A_\pi$ implies $(h_{s_1}(a_1), \ldots, h_{s_n}(a_n)) \in B_\pi]$. Models and model morphisms form a category denoted $Mod(\Sigma)$ too (just like the class of models), with composition defined as sort-wise function composition. By a *surjective* (*injective*) morphism we mean a morphism which is surjective (injective) on each sort. Because of the weak form of commutation imposed on morphisms w.r.t. the relational part of models, relations and functions do not behave similarly along arbitrary morphisms, but only along closed ones: a morphism $h : A \to B$ is called *closed* if the relation preservation condition holds in the "iff" form, that is, $(a_1, \ldots, a_n) \in A_\pi$ iff $(h_{s_1}(a_1), \ldots, h_{s_n}(a_n)) \in B_\pi$. A morphism $h : A \to B$ is called *strong* if the target relations are covered through h by the source relation, that is, for each $\pi \in P_w$ with $w = s_1 \ldots s_n$ and $(b_1, \ldots, b_n) \in B_\pi$, there exists $(a_1, \ldots, a_n) \in A_\pi$ such that $(h_{s_1}(a_1), \ldots, h_{s_n}(a_n)) = (b_1, \ldots, b_n)$. Closed injective morphisms and strong surjective morphisms naturally capture the notions of embedding and homomorphic image respectively. Note that for surjective morphisms, "closed" is equivalent to "strong", but we shall keep the standard terminology using the phrase "strong surjective morphism".

If $\phi : (S, F, P) \to (S', F', P')$ is a signature morphism, for each Σ-morphism $h' : A' \to B'$, its ϕ-reduct is the Σ-morphism $h' {\restriction_\phi}\colon A' {\restriction_\phi} \to B' {\restriction_\phi}$ defined by $(h {\restriction_\phi})_s = h_{\phi(s)}$ for each $s \in S$. The mapping $h' \mapsto h' {\restriction_\phi}$ extends $Mod(\phi)$ to a functor between the categories of Σ'-models and Σ-models.

For a fixed first-order signature $\Sigma = (S, F, P)$, we next define some sets of, possibly infinitary (in the sense of admitting conjunctions and/or disjunctions on arbitrary sets of sentences), first-order sentences. Whenever we do not explicitly state otherwise, the sentences are considered finitary:
- $\mathcal{FO}$, the set of first-order sentences;
- $\mathcal{P}os$, the set of *positive* sentences, that is, constructed inductively from atomic formulae by means of any first-order constructs, except negation;
- $\forall$, the set of sentences of the form $(\forall x_1, x_2, \ldots, x_k)e$, where $k \in \mathbb{N}$ and e is a quantifier free formula;
- $\exists$, the set of sentences of the form $(\exists x_1, x_2, \ldots, x_k)e$, where $k \in \mathbb{N}$ and e is a quantifier free formula;
- $\mathcal{UH}$, the set of *universal Horn clauses*, that is, sentences of the form $(\forall x_1, x_2, \ldots, x_k)(e_1 \wedge e_2 \ldots \wedge e_p) \Rightarrow e$, with $k, p \in \mathbb{N}$ and e_i, e atomic formulae;
- $\mathcal{UA}$, the set of *universal atoms*, that is, sentences of the form $(\forall x_1, x_2, \ldots x_k)e$, where $k \in \mathbb{N}$ and e is an atomic formula;
- $\Box$, the set of sentences of the form $(\exists x_1)(\forall y_1^1, y_1^2, \ldots, y_1^{p_1}) \ldots (\exists x_k)(\forall y_k^1, y_k^2, \ldots, y_k^{p_k}) \bigwedge_{u=1}^{r} \bigvee_{v=1}^{s_u} e_{u,v}$, where $k, r, p_i, s_u \in \mathbb{N}$, and each $e_{u,v}$ is either atomic, or of the form $\neg\sigma(y_i^1, \ldots, y_i^{p_i-1}) = y_{p_i}^i$, or of the form $\neg\pi(y_i^1, \ldots, y_i^{p_i})$ (this strange looking set of sentences is closely related to the class of strong surjective morphisms - see Proposition 7);
- $\forall\vee$, the set of universally quantified disjunctions of atoms, i.e., sentences of the form $(\forall x_1, x_2, \ldots, x_k)(e_1 \vee e_2 \ldots \vee e_p)$ where $p \in \mathbb{N}$ and e_i are atomic formulae;
- $\mathcal{FO}_\infty$, $\mathcal{UH}_\infty$, $\forall\vee_\infty$, the infinitary extensions of $\mathcal{FO}$, $\mathcal{UH}$, $\forall\vee$, respectively; in the former case, infinite conjunction and disjunction is allowed as a new rule of constructing sentences; in the latter two cases, $e_1 \wedge e_2 \ldots \wedge e_p$ and $e_1 \vee e_2 \ldots \vee e_p$ are replaced by any possibly infinite sentence- conjunction and disjunction respectively.

Let us next consider the binary relations S, Ext, H, Hs, Ech, Upw, Ur on Σ-models:
- $A\ S\ B$ iff there exists a strong injective morphism between B and A (in other words, iff B is isomorphic to a submodel of A);
- $A\ Ext\ B$ iff $B\ S\ A$, that is, iff B is isomorphic to an extension of A;
- $A\ H\ B$ iff there exists a surjective morphism between A and B;
- $A\ Hs\ B$ iff there exists a strong surjective morphism between A and B;
- $A\ Ech\ B$ iff A is elementary equivalent to B;
- $A\ Upw\ B$ iff B is isomorphic to an ultrapower of A;
- $A\ Ur\ B$ iff B is an ultraradical of a model isomorphic to A (note that $Ur = Upw^{-1}$).

Recall that any binary relation on $Mod(\Sigma)$, in particular the ones above, has an associated operator (bearing the same name) on $Mod(\Sigma)$. Besides these, we shall also consider the operators P, Fp, Up and Ur on $Mod(\Sigma)$ defined below:
- $P(\mathcal{M}) = \mathcal{M} \cup \{$all direct products of models in $\mathcal{M}\}$;
- $Fp(\mathcal{M}) = \mathcal{M} \cup \{$all filtered products of models in $\mathcal{M}\}$;
- $Up(\mathcal{M}) = \mathcal{M} \cup \{$all ultraproducts of models in $\mathcal{M}\}$;

All these constructions are considered up to isomorphism; for instance, the operator *Up* "grabs" into the class not only the ultraproducts standardly constructed as quotients of direct products, but all models isomorphic to them.

We are not going to recall filtered products and powers here - the reader is referred to [7, 24]. As a matter of notation, if $\mathcal{F} \subseteq \mathcal{P}(I)$ is a filter, A a model, and $(A_i)_{i\in I}$ a family of models, then $\prod_{\mathcal{F}} A_i$ and $A^I/_{\mathcal{F}}$ denote the filtered product of $(A_i)_i$ over $\mathcal{F}$ and the filtered power of A over $\mathcal{F}$, respectively. If $\mathcal{F}$ is an ultrafilter then $\prod_{\mathcal{F}} A_i$ is called an ultraproduct of $(A_i)_i$, $A^I/_{\mathcal{F}}$ is called an ultrapower of A, and A is called an ultraradical of $A^I/_{\mathcal{F}}$.

The next proposition lists some known axiomatizability results. For details, the reader is referred to [7] (Section 5.2), [24] (Sections 25 and 26), [1], [26], and [10].

Proposition 7. *Let $\Sigma = (S, F, P)$ be a signature and $\mathcal{M} \subseteq Mod(\Sigma)$ be a class of models. If the pair (T, Ops), consisting of a type T of Σ-sentences and a set Ops of operators on $Mod(\Sigma)$, is one of: $(\mathcal{FO}, \{Up, Ur\})$, $(\mathcal{P}os, \{Up, Ur, H\})$, $(\forall, \{S, Up\})$, $(\exists, \{Ext, Up, Ur\})$, $(\mathcal{UH}, \{S, Fp\})$, $(\mathcal{UA}, \{S, H, P\})$, $(\forall\vee, \{Hs, S, Up\})$, $(\Box, \{Hs, Up\})$, $(\mathcal{UH}_\infty, \{S, P\})$, $(\forall\vee_\infty, \{Hs, S\})$*

then the following are equivalent:

1. $\mathcal{M}$ is of the form Γ^*, with $\Gamma \subseteq T$,
2. $\mathcal{M}$ is a fixed point of all the operators in the set Ops.

Consider the following properties for a morphism $\phi : (S, F, P) \to (S', F', P')$.

(IS) ϕ is **injective on sorts**, i.e., ϕ^{st} is injective;
(IR) ϕ is **injective on relation symbols**, i.e., ϕ^{rl} is injective;
(I) ϕ is **injective** on sorts, operation- and relation- symbols, i.e., ϕ^{st}, ϕ^{op}, and ϕ^{rl} are injective;
(RS) there are no operation symbols in $F' \setminus \phi(F)$, having the *result sort* in $\phi(S)$.

Proposition 8. *For each morphism $\phi : \Sigma = (S, F, P) \to \Sigma' = (S', F', P')$,*

1. *$Mod(\phi)$ preserves fixed points of P, Fp, Up;*
2. (I) *implies that $Mod(\phi)$ lifts S, H, Hs and preserves fixed points of Ext.*
3. (IS) *and* (RS) *imply that $Mod(\phi)$ preserves fixed points of S, Hs, and lifts Ext;*
4. (IS), (IR) *and* (RS) *imply that $Mod(\phi)$ preserves fixed points of H;*
5. (IS) *implies that $Mod(\phi)$ lifts Ur;*

Proof. Throughout this proof, for any signature Σ, Σ-morphism $h : A \to B$, and $w = s_1 \ldots s_n \in S^*$, $h_w : A^w \to B^w$ denotes the mapping defined by $h_w(a_1, \ldots, a_n) = (h_{s_1}(a_1), \ldots, h_{s_n}(a_n))$.
1: Follows from the well-known facts that $Mod(\phi)$ preserves direct products and filtered colimits and that filtered products are filtered colimits of direct products.
2: Proven in [10], Proposition 1. Note that for a binary relation R, $Mod(\phi)$ lifts R iff ϕ lifts R^{-1} according to the terminology in [10]; also, our relations S, H and Hs coincide with the inverses of the relations $\stackrel{\mathcal{S}_c}{\to}$, $\stackrel{\mathcal{H}_r}{\leftarrow}$ and $\stackrel{\mathcal{H}_s}{\leftarrow}$ defined in [10], respectively.

3 and 4: Let A' be a Σ'-model and B a Σ-model.

Preservation of fixed points of S and lifting of Ext: Suppose there exists a strong injective morphism $i : B \to A'\lceil_\phi$. Let B' be the following Σ'-model:
- For each $s' \in S'$, let $B'_{s'} = B_s$ if s' has the form $\phi^{st}(s)$ and $B'_{s'} = A'_{s'}$ otherwise. Since ϕ^{st} is injective, the definition is not ambiguous. We can now define for each $s' \in S'$, $i'_{s'} : B'_{s'} \to A'_{s'}$ to be i_s if s' has the form $\phi^{st}(s)$ and $1_{A'_{s'}}$ otherwise;
- For each $\sigma' \in F'_{w'\to s'}$, let $B'_{\sigma'} = B_\sigma$ if σ' has the form $\phi^{op}_{w\to s}(\sigma)$ and $B'_{\sigma'}(\overline{b'}) = A'_{\sigma'}(i'_{w'}(\overline{b'}))$ for each $\overline{b'} \in B'^{w'}$ otherwise. (Note that, because of (RS), in the latter case of the definition $s' \notin \phi^{st}(S)$, thus $A'_{\sigma'}(i'_{w'}(\overline{b'})) \in B_{s'}$.) Let us show that the definition above is not ambiguous. Consider $\sigma_1, \sigma_2 \in F_{w\to s}$ such that $\phi^{op}_{w\to s}(\sigma_1) = \phi^{op}_{w\to s}(\sigma_2)$. Then $(A'\lceil_\phi)_{\sigma_1} = (A'\lceil_\phi)_{\sigma_2}$ and since i is injective it follows that $B_{\sigma_1} = B_{\sigma_2}$.
- For each $\pi' \in P'_{w'}$, let $B'_{\pi'} = (i'_{w'})^{-1}(A'_{\pi'})$.

Thus, B' is a Σ'-model and i' is an injective morphism. Furthermore, i' is strong from the way the relations $B'_{\pi'}$ were defined on B'. Also, the models $B'\lceil_\phi$ and B have the same sort carriers and operations by the definition of B'. Finally, for any $\pi \in P_w$, we have that $B_\pi = (i_w)^{-1}((A'\lceil_\phi)_\pi) = (i'_{\phi^{st}(w)})^{-1}(A'_{\phi^{rl}(\pi)}) = B'_{\phi^{rl}(\pi)} = (B'\lceil_\phi)_\pi$, hence $B'\lceil_\phi$ and B coincide on the relational part too.

Preservation of fixed points of H and Hs: Suppose there exists a surjection $h : A'\lceil_\phi \to B$. Let B' be the following Σ'-model:
- For each $s' \in S'$, let $B'_{s'} = B_s$ if s' has the form $\phi^{st}(s)$ and $B'_{s'} = \{\star\}$ (a singleton) otherwise. Since ϕ^{st} is injective, the definition is not ambiguous. We now define for each $s' \in S'$, $h'_{s'} : A'_{s'} \to B'_{s'}$ to be h_s if s' has the form $\phi^{st}(s)$ and the only possible mapping otherwise;
- For each $\sigma' \in F'_{w'\to s'}$, let $B'_{\sigma'} = B_\sigma$ if σ' has the form $\phi^{op}_{w\to s}(\sigma)$ and $B'_{\sigma'}(\overline{b'}) = \star$ for each $\overline{b'} \in B'_{w'}$ otherwise. (Note that, because of (RS), in the latter case of the definition s' does not have the form $\phi^{st}(s)$, thus $B'_{s'} = \{\star\}$.) Let us show the definition above is not ambiguous. Consider $\sigma_1, \sigma_2 \in F_{w\to s}$ such that $\phi^{op}_{w\to s}(\sigma_1) = \phi^{op}_{w\to s}(\sigma_2)$. Then $(A'\lceil_\phi)_{\sigma_1} = (A'\lceil_\phi)_{\sigma_2}$ and, since h is surjective, it follows that $B_{\sigma_1} = B_{\sigma_2}$.
- Let $\pi' \in P'_{w'}$. If h is strong, let $B'_{\pi'} = h'_{w'}(A'_{\pi'})$. If h is not strong (thus we work under the hypothesis that ϕ^{rl} is injective), let $B'_{\pi'} = B_\pi$ if π' has the form $\phi^{rl}(\pi)$ and $B'_{\pi'} = B'^{w'}$ otherwise.

Thus, B' is a Σ'-model and h' is a surjective morphism. Furthermore, the models $B'\lceil_\phi$ and B have the same sort carriers and operations by the definition of B'. If h is not strong, then $B'\lceil_\phi$ and B coincide on the relational part too, by the definition of B'. On the other hand, if h is strong, then for any $\pi \in P_w$ we have that $B'_{\pi'} = h_{w'}(A'_{\pi'})$, hence $B_\pi = h_w((A'\lceil_\phi)_\pi) = h'_{\phi^{st}(w)}(A'_{\phi^{rl}(\pi)}) = B'_{\phi^{rl}(\pi)} = (B'\lceil_\phi)_\pi$. Note that in case h is strong, h' is strong too.

5: Let A' be a Σ'-model and let B be a Σ-model isomorphic to an ultrapower of $A'\lceil_\phi$, say $A'\lceil_\phi{}^I/_{\mathcal{F}}$. Let $C' = A'^I/_{\mathcal{F}}$. It is known [7] that $C'\lceil_\phi$ is equal to $A'\lceil_\phi{}^I/_{\mathcal{F}}$, hence is isomorphic to B. Since ϕ has (IS), it is easy to define a Σ'-model B' such that $B'\lceil_\phi = B$ and B' is isomorphic to C', whence $B'\, Ur\, A'$. $\square$

The table below states interpolation results for diverse types of sentences. It should be read as: given a weak amalgamation square of signatures as in

Definition 2 and $\Gamma_1 \subseteq Mod(\Sigma_1)$, $\Gamma_2 \subseteq Mod(\Sigma_2)$, if Γ_1 and Γ_2 are sentences of the indicated types such that $\phi_2'(\Gamma_1) \models \phi_1'(\Gamma_2)$, then there exists an interpolant Γ for them, of the indicated type; the semantic conditions under which this situation holds are given in the $Mod(\phi_1)$- and $Mod(\phi_2)$- columns of the table, with the meaning that $Mod(\phi_1)$ *preserves* fixed points of the indicated operator and $Mod(\phi_2)$ *lifts* the indicated operator; Id stands for the identity operator. These semantic conditions are implied by the concrete syntactic conditions listed in the ϕ_1- and ϕ_2- columns of the table; "any" means that no restriction is posed on the signature morphism.

	Γ_1	Γ_2	Γ	$Mod(\phi_1)$	$Mod(\phi_2)$	ϕ_1	ϕ_2
1	$\mathcal{FO}$	$\mathcal{FO}$	$\mathcal{FO}$	Up	Ur	any	(IS)
2	$\mathcal{FO}$	$\mathcal{P}os$	$\mathcal{P}os$	Up	$H\,;Ur$	any	(I)
3	$\mathcal{P}os$	$\mathcal{FO}$	$\mathcal{P}os$	$Up\,;H$	Ur	(IS), (IR), (RS)	(IS)
4	$\mathcal{FO}$	$\forall$	$\forall$	Up	S	any	(I)
5	$\forall$	$\mathcal{FO}$	$\forall$	$Up\,;S$	Id	(IS), (RS)	any
6	$\forall$	$\mathcal{P}os$	$\forall\vee$	$Up\,;S$	Hs	(IS), (RS)	(I)
7	$\mathcal{FO}$	$\exists$	$\exists$	Up	$Ext;Ur$	any	(IS), (RS)
8	$\exists$	$\mathcal{FO}$	$\exists$	$Up\,;Ext$	Ur	(I)	(IS)
9	$\mathcal{FO}$	$\mathcal{UH}$	$\mathcal{UH}$	Fp	S	any	(I)
10	$\mathcal{UH}$	$\mathcal{FO}$	$\mathcal{UH}$	$Fp\,;S$	Id	(IS), (RS)	any
11	$\mathcal{UH}$	$\mathcal{UA}$	$\mathcal{UA}$	P	$S\,;H$	any	(I)
12	$\mathcal{UA}$	$\mathcal{FO}$	$\mathcal{UA}$	$P\,;S\,;H$	Id	(IS), (IR), (RS)	any
13	$\mathcal{UH}$	$\mathcal{P}os$	$\mathcal{UA}$	$P\,;S$	H	(IS),(RS)	(I)
14	$\mathcal{FO}$	$\forall\vee$	$\forall\vee$	Up	$S\,;Hs$	any	(I)
15	$\forall\vee$	$\mathcal{FO}$	$\forall\vee$	$Up\,;S\,;Hs$	Id	(IS), (RS)	any
16	$\mathcal{FO}$	$\square$	$\square$	Up	Hs	any	(I)
17	$\square$	$\mathcal{FO}$	$\square$	$Up\,;Hs$	Id	(IS), (RS)	any
18	$\mathcal{UH}_\infty$	$\mathcal{UA}$	$\mathcal{UA}$	P	$S\,;H$	any	(I)
19	$\mathcal{UH}_\infty$	$\mathcal{FO}_\infty$	$\mathcal{UH}_\infty$	$P\,;S$	Id	(IS), (RS)	any
20	$\mathcal{FO}_\infty$	$\forall\vee_\infty$	$\forall\vee_\infty$	Id	$S\,;Hs$	any	(I)
21	$\forall\vee_\infty$	$\mathcal{FO}_\infty$	$\forall\vee_\infty$	$S\,;Hs$	Id	(IS), (RS)	any
22	$\mathcal{FO}$	$\forall\vee_\infty$	$\forall\vee$	Up	$S\,;Hs$	any	(I)

Theorem 2. *The results stated in the above table hold, i.e., in each of the 22 cases, if ϕ_1 and ϕ_2 satisfy the indicated properties, Γ_1 and Γ_2 have the indicated types and $\phi_2'(\Gamma_1) \models \phi_1'(\Gamma_2)$, then there exists an interpolant Γ of the indicated type.*

Proof. Let F and G be the operators in the $Mod(\phi_1)$- and $Mod(\phi_2)$- column respectively.

We are going to apply Theorem 1. First, we check hypotheses 3 and 4 of this theorem, i.e., prove that if ϕ_1 and ϕ_2 are as indicated then $Mod(\phi_1)$ preserves fixed points of F and $Mod(\phi_2)$ lifts G. By Proposition 8, we get that $Mod(\phi_1)$ preserves fixed points of, and $Mod(\phi_2)$ lifts, all the atomic components of F and G respectively. (For instance, in line 12, $Mod(\phi_2)$ lifts all the atomic components of $P\ ;S\ ;H$, i.e., P, S, and H.) Moreover, since all the involved operators are monotone, we can apply Proposition 6.1 to obtain that $Mod(\phi_2)$ lifts G. In order to apply Proposition 6.2 for $Mod(\phi_1)$, we further need that, if F_0, F_1 and F_2 are operators such that F is listed as $F_1\ ;F_2$ or as $F_0\ ;F_1\ ;F_2$ in the $Mod(\phi_1)$-column, then $F_1\ ;F_2\ ;F_1 = F_1\ ;F_2$ or $[F_1\ ;F_2\ ;F_1 = F_1\ ;F_2$ and $F_0\ ;(F_1\ ;F_2)\ ;F_0 = F_0\ ;(F_1\ ;F_2)]$ respectively. According to Proposition 6.3, it would suffice that each of F_0, F_1, F_2, $F_1\ ;F_2$, $F_0\ ;F_1\ ;F_2$ be closure operators. We are going to show even more: that both F and G, as well all their components, are closure operators. (The components need not be atomic: for instance, the components of $P\ ;S\ ;H$ in the table cell of line 12 and column $Mod(\phi_1)$ are P, S, H, $P\ ;S$, $S\ ;H$, and $P\ ;S\ ;H$ itself.) Well-known closure operators are P, S, H, Hs (obviously), Up, Ur (see [7]), $P\ ;H\ ;S$, $Fp\ ;S$, $P\ ;S$ (the famous closure operators of Birkhoff and Malçev, the latter for both finitary and infinitary Horn clauses - see [1]), $Up\ ;S\ ;Hs$ (see [1, 10]). Moreover, $Up\ ;H$, $Up\ ;Hs$, $Up\ ;S$, $Up\ ;Ext$, $H\ ;Ur$, $S\ ;Ur$, $Ext\ ;Ur$, $S\ ;H$, and $S\ ;Hs$ are closure operators because their components are closure operators and because $H\ ;Up \sqsubseteq Up\ ;H$, $Hs\ ;Up \sqsubseteq Up\ ;Hs$, $S\ ;Up \sqsubseteq Up\ ;S$, $Ext\ ;Up \sqsubseteq Up\ ;Ext$, $Ur\ ;H \sqsubseteq H\ ;Ur$, $Ur\ ;S \sqsubseteq S\ ;Ur$, $Ur\ ;Ext \sqsubseteq Ext\ ;Ur$, $H\ ;S \sqsubseteq S\ ;H$, and $Hs\ ;S \sqsubseteq S\ ;Hs$. Indeed, most of these inclusions are easily seen to hold. We only check $H\ ;Up \sqsubseteq Up\ ;H$, $Hs\ ;Up \sqsubseteq Up\ ;Hs$, $H\ ;S \sqsubseteq S\ ;H$, and $Hs\ ;S \sqsubseteq S\ ;Hs$. The first two equalities follow from the fact that, for any family (strong) surjective morphisms $(h_i : A_i \to B_i)_{i\in I}$ and any ultrafilter $\mathcal{F}$ on I, the induced mapping $h : A^I/_{\mathcal{F}} \to B^I/_{\mathcal{F}}$ (defined by $h((a_i)_i/\mathcal{F}) = (h_i(a_i)_i)/\mathcal{F})$ is also a (strong) surjective morphism. The last two equalities follow from the fact that if $h : B \to C$ is a (strong) surjective morphism and A is a submodel of C, then $h^{-1}(A)$, which is a submodel of B with induced operations and relations, yields a restriction-corestriction of h to $h^{-1}(A) \to A$ which is also a (strong) surjective morphism.

Note that we also checked hypothesis 2, since we proved all the involved operators to be closure operators. We now check hypothesis 1. Because both F and G are closure operators, in order that $F\ ;G\ ;F = F\ ;G$, it suffices that $G\ ;F \sqsubseteq F\ ;G$. So we check $G\ ;F \sqsubseteq F\ ;G$. Most of the needed inclusions were already discussed. We mention only the non-obvious ones that are left.

- $Ur\ ;Up \sqsubseteq Up\ ;Ur$ follows from the Keisler-Shelah theorem [24], which says that $Ech \sqsubseteq Upw\ ;Ur$, together with the easy fact that $Ech\ ;Up \sqsubseteq Up\ ;Ech$; indeed, we have that $Ur\ ;Up \sqsubseteq Ech\ ;Up \sqsubseteq Up\ ;Ech \sqsubseteq Up\ ;Upw\ ;Ur \sqsubseteq Up\ ;Up\ ;Ur = Up\ ;Ur$;
- $(Ext\ ;Ur)\ ;Up \sqsubseteq Up\ ;(Ext\ ;Ur)$ follows from $Ur\ ;Up \sqsubseteq Up\ ;Ur$ and $Ext;\ Up \sqsubseteq Up\ ;Ext$;
- $Ur\ ;(Up\ ;Ext) \sqsubseteq (Up\ ;Ext)\ ;Ur$ follows from $Ur\ ;Up \sqsubseteq Up\ ;Ur$ and $Ur\ ;Ext \sqsubseteq Ext;\ Ur$.

Thus the hypotheses of Theorem 1 are fulfilled. We obtained that Γ is closed under both F and G.[4] In order to be able to apply Proposition 7 to Γ, it still

[4] Strictly speaking, F and G for Γ are different to the operators F for Γ_1 and G for Γ_2; however, there is no danger in overlapping the notations here.

remains to prove that any class of models which is a fixed point of F (of G) is also a fixed point of all the operators that compose F (that compose G). This easily follows from the pointed fact that both F, G, and all their components are closure operators. Indeed, for instance assume that F is listed in the table as $F_0 \,; F_1 \,; F_2$, and let $\mathcal{M}$ be a fixed point of F; then $F_1(\mathcal{M}) = F_1(F(\mathcal{M})) \subseteq F(F(\mathcal{M})) = F(\mathcal{M})$, hence $\mathcal{M}$ is a fixed point of F_1 too. $\square$

Let us analyze the results listed in the table above. A first thing to notice are the syntactic conditions on signature morphisms, in many cases weaker than, or equal to, full injectivity. In fact, if we consider only relational languages, i.e., without operation symbols, *all* the conditions are so. As for operation symbols, condition (RS), requiring that no new operations be added on old sorts, expresses precisely the principle of data encapsulation in algebraic terms [18]. It is interesting (at least for the authors) that this condition, stated in the table on the signature morphism ϕ_1 because of purely technical reasons, turns out to be related to information hiding, and furthermore to the fact that in algebraic specifications interpolation is used with ϕ_1 as the *hiding morphism* - see Section 5. As pointed out by the examples in Section 2, it seems that the degree of generality that one can allow on signature morphism increasses with the expressive power of a logic. For instance, line 1 says that first-order interpolation holds whenever the righthand morphism is injective on sorts (and, in fact, since in full first-order logic Craig interpolation is equivalent to the symmetrical property of Robinson consistency,[5] *either one* of the morphisms being injective on sorts would do.). On the other hand, universal Horn clauses (lines 9 and 10), and then universal atoms (lines 11, 12, 13) require stronger and stronger assumptions on the signature morphisms. Second, all the results say more than interpolation within a certain type T of sentences, ensuring that the interpolant has type T provided *one* of the starting sets has type T. Particularly interesting results are listed in lines 6, 13, and 22. Here, the interpolant inherits properties from both sides, strictly "improving" the type of both sides. Finally, note that the case of existential sentences (lines 7 and 8), is neither dual, nor reducible, to the case of universal sentences (lines 5 and 6); in fact, one could see that the results differ and are not mutually symmetric. Indeed, while within previously asumed *elementary* classes, the correspondances $\forall$-submodels and $\exists$-extensions are dual, the symmetry is broken when one invokes ultraradicals, which are particular cases of submodels, but of course not of extensions.

Regarding finiteness of Γ, as noted in [10], it is easy to see that if Γ_2 is finite, by compactness of first-order logic, Γ can be chosen to be finite too in our cases of finitary sub-first-order logics. On the other hand, the finiteness of Γ_1 does not necessarily imply the finite axiomatizability of Γ^*. Indeed, assume that $\Sigma = \Sigma_2 \subseteq \Sigma_1$, ϕ_1 is the inclusion of signatures, and ϕ_2 the identity. Then $\Sigma' = \Sigma_1$, $\phi_1' = \phi_1$, and $\phi_2' = \phi_2$. Thus the finite interpolation problem comes to the following: assuming $\Gamma_1 \models \Gamma_2$ in Σ_1, find a finite $\Gamma \subseteq \mathit{Sen}(\Sigma)$ such that $\Gamma_1 \models \Gamma \models \Gamma_2$; in other words, prove that there exists a finite subset Δ_1 of $\Gamma_1^\bullet$

[5] This is not true however for our examples of sub-first-order logic.

consisting of Σ-sentences such that $\Delta_1 \models \Gamma_2$ in Σ. But this cannot be always achieved, as shown by the case where Γ_2 is a Σ-theory ($\bullet$-closed set of sentences), finitely axiomatizable over the extended signature Σ_1 by Γ_1, but not finitely axiomatizable over Σ. (Such a theory is known to exist by a famous theorem of Kleene.) In our model-theoretical approach, the impossibility of relating the finiteness of Γ_1 to that of Γ is illustrated by the fact that the operator of taking ultraproduct components (classically related to finite axiomatizability [7]) is not preserved by reduct functors, but lifted by them.

Interpolation in second-order and higher-order logic

We shall next apply Theorem 1 to prove interpolation for the unsorted versions of second-order and higher-order logics. Interpolation is of course known to hold for these logics, in case of inclusions of signatures. Our generalizations to arbitrary signature morphisms emphasize once more how easy does interpolation follow in sufficiently expressive logics; recall that expression power was the main obstacle to interpolation in our examples in Section 2.

In second-order logic, signatures and models are the same as in unsorted first-order logic,[6] but the sentences are extended by allowing quantification not only over variables denoting elements on sorts, but also over variables denoting operations and relations of arbitrary arity. Given a morphism of signatures $\phi : \Sigma \to \Sigma'$, a Σ'-sentence e and a Σ-model A, we define $A \models (\forall\phi)e'$ by: for each Σ'-model A' such that $A'\upharpoonright_\phi = A$, it holds that $A' \models e'$.

Proposition 9. *Consider a weak amalgamation square of signatures as in Definition 2, such that the signatures are finite (in the sense that their components F and P are finite). Let Γ_1 and Γ_2 be two sets of second-order Σ_1- and Σ_2- sentences respectively. Then there exists a set Γ of second-order Σ-sentences which is an interpolant for Γ_1 and Γ_2.*

Proof. The key point in the proof is to notice that second-order logic admits universal quantification over ϕ_2, in the sense that, for each Σ_2-sentence e_2, there exists a Σ-sentence e such that e is semantically equivalent to $(\forall\phi_2)e_2$. Indeed, given a Σ_2-sentence e_2, $(\forall\phi_2)e_2$ is equivalent to the Σ-sentence $(\forall X)(d_1 \wedge \ldots \wedge d_n) \Rightarrow e$, where X contains all the Σ_2-symbols (i.e., relation- and operation-symbols) that are not ϕ-images of Σ-symbols, and $d_1, \ldots d_n$ are all the equations of the form $u_1 = u_2$, where u_1 and u_2 are Σ-symbols such that $\phi_2(u_1) = \phi_2(u_2)$. Thus we can refer to such sentences quantified over ϕ_2 as belonging to $Sen(\Sigma)$.

Now, using Theorem 1 for $\mathcal{U} = Id$ and $\mathcal{V} = Mod(\phi_2)$, it suffices to show that $Mod(\phi_2)$ lifts the second-order axiomatizability hull operator $\bullet : \mathcal{P}(Mod(\Sigma)) \to \mathcal{P}(Mod(\Sigma))$. Let $\mathcal{M} \subseteq Mod(\Sigma)$. We need to show that $Mod(\phi_2)^{-1}(\mathcal{M}^\bullet) \subseteq Mod(\phi_2)^{-1}(\mathcal{M})^\bullet$. For this, let $A_2 \in Mod(\phi_2)^{-1}(\mathcal{M}^\bullet)$. Then $A_2\upharpoonright_{\phi_2} \in \mathcal{M}^\bullet$. In order to prove $A_2 \in Mod(\phi_2)^{-1}(\mathcal{M})^\bullet$, let e_2 be a second-order Σ-sentence such that $Mod(\phi_2)^{-1}(\mathcal{M}) \models e_2$. Then, for any $B_2 \in Mod(\Sigma_2)$ such that $B_2\upharpoonright_{\phi_2} \in \mathcal{M}$,

[6] That is, signatures have only one sort, hence we can omit it and regard the signatures as pairs (F, P).

it is the case that $B_2 \models e_2$; but this precisely means that $\mathcal{M} \models (\forall\phi_2)e_2$, hence $A_2{\restriction_{\phi_2}} \models (\forall\phi_2)e_2$. From this latter fact and the definition of ϕ_2-quantification, one can deduce $A_2 \models e_2$, which is what we needed. □

Let us now sketch the formal setting of *unsorted higher-order logic*. Let b be a fixed symbol, which will stand for the *basic type*. The set T of types is defined recursively by the following rules: 1) $b \in T$; 2) if $t_1, t_2 \in T$, then $t_1 \to t_2$ and $t_1 \times t_2 \in T$. A *higher-order signature* is a T-indexed set $\Sigma = (\Sigma_t)_{t\in T}$. A morphism between Σ and Σ' is a T-indexed mapping $\phi = (\phi_t)_{t\in T}$, where $\phi_t : \Sigma_t \to \Sigma'_t$ for all $t \in T$. To each set D, one can naturally associate the T-indexed set $(D_t)_{t\in T}$ as follows: $D_b = D$, $D_{t_1\to t_2} = [D_{t_1} \to D_{t_2}]$ (the set of functions between D_{t_1} and D_{t_2}), $D_{t_1\times t_2} = D_{t_1} \times D_{t_2}$. A Σ-model is structure of the form $(A, (A_t(\sigma))_{t\in T,\sigma\in\Sigma_t})$, where $A_t(\sigma) \in A_t$ for each $t \in T$ and $\sigma \in \Sigma_t$. The Σ-sentences are just the first-order sentences associated with the many-sorted first-order signature $(T, F, \emptyset)$, where $F_{w,t} = \emptyset$ if $w \neq \lambda$ and $F_{\lambda,t} = \Sigma_t$ for each $t \in T$. The satisfaction relation, as well as the mappings on sentences and models associated with signature morphisms are the obvious ones.

Proposition 10. *Consider a weak amalgamation square of signatures as in Definition 2, just that the signatures and signature morphisms are assumed to be higher-order ones. Assume that all the signatures are finite, in the sense that all their components Σ_t are finite. Let Γ_1 and Γ_2 be two sets of Σ_1- and Σ_2-sentences respectively. Then there exists a set Γ of Σ-sentences which is an interpolant for Γ_1 and Γ_2.*

Proof. Similar to the proof of 9, using Theorem 1 together with the fact that higher-order logic admits quantification over ϕ_2. □

As a general rule, it seems that in any "reasonable" logical system (such as the ones formalizable as institutions [19]), the Craig interpolation property[7] holds for each weak amalgamation square for which the logic admits quantification over the righthand morphism ϕ_2.

5 Applications to Formal Specification

Next we consider some applications of our results to formal specification and module algebra. We shall provide enough technical details in order to ensure the clear understanding of these (rather technical) applications.

A *flat module* in specification languages can be viewed as a pair consisting of a signature Σ and a finite set of sentences Γ describing the class of admissible Σ-models/implementations, Γ^*. One is usually interested in the set $\Gamma^\bullet$ of *all* properties satisfied by the admissible models. Thus one initially starts with flat theories, i.e., pairs (Σ, Γ), where Γ is a $\bullet$-closed set of Σ-sentences. In practice, one might additionally require that Γ be finitely or recursively presented. According to [4], the semantics of modules by their theories is a strong candidate for becoming the standard module algebra semantics.

[7] See also [32, 10] for for more about institutional Craig Interpolation.

Diverse operations are used to build up structured theories out of flat ones, among which the *export* and *combination* operators [4], $\Box$ and $+$.[8] They are defined as follows, for each signature Σ' and theory (Σ, Γ), let $\Sigma' \Box (\Sigma, \Gamma)$ be $(\Sigma' \cap \Sigma, \iota^{-1}(\Gamma))$, where $\Sigma' \cap \Sigma \overset{\iota}{\hookrightarrow} \Sigma$ and for theories (Σ_1, Γ_1) and (Σ_2, Γ_2), let $(\Sigma_1, \Gamma_1) + (\Sigma_2, \Gamma_2)$ be $(\Sigma_1 \cup \Sigma_2, (\Gamma_1 \cup \Gamma_2)^\bullet)$, where intersections and unions of signatures are defined standardly in a component-wise fashion.

$\Box$ restricts the interface of the theory (Σ, Γ) to common symbols of Σ' and Σ, while $+$ just puts together two theories in their union signature. A very desirable property of specification frameworks is the following restricted distributivity law:

$$\Sigma' \Box ((\Sigma_1, \Gamma_1) + (\Sigma_2, \emptyset^\bullet)) = (\Sigma' \Box (\Sigma_1, \Gamma_1)) + (\Sigma' \cap \Sigma_2, \emptyset^\bullet)$$

As discussed in [4, 15], one could not usually count on full distributivity. It is shown in [4] that, if one works within first-order logic, restricted distributivity holds, being implied by common union-intersection interpolation. Their proof, as many others in the mentioned paper, is rather logic-independent, so it works for any logic that has first-order signatures and satisfies interpolation. In particular, it works for all the sublogics of (finitary or infinitary) first-order logic appearing in the table that precedes Theorem 2. For completeness, we restate the proof here, translated into our language.

The two members of the desired equality are, by the definition of $\Box$ and $+$, $(\Sigma' \cap (\Sigma_1 \cup \Sigma_2), j^{-1}(k(\Gamma_1)^\bullet))$, and $((\Sigma' \cap \Sigma_1) \cup (\Sigma' \cap \Sigma_2), p(i^{-1}(\Gamma_1))^\bullet)$ respectively, where i, j, k, p are the signature inclusions $\Sigma' \cap \Sigma_1 \hookrightarrow \Sigma_1$, $\Sigma' \cap (\Sigma_1 \cup \Sigma_2) \hookrightarrow \Sigma_1 \cup \Sigma_2$, $\Sigma_1 \hookrightarrow \Sigma_1 \cup \Sigma_2$, $\Sigma' \cap \Sigma_1 \hookrightarrow \Sigma' \cap (\Sigma_1 \cup \Sigma_2)$ respectively. (These signature inclusions needed to be explicitly denoted in order to make clear each time within which signature the bullet operator is applied.) Since $\Sigma' \cap (\Sigma_1 \cup \Sigma_2) = (\Sigma' \cap \Sigma_1) \cup (\Sigma' \cap \Sigma_2)$, it remains to prove $j^{-1}(k(\Gamma_1)^\bullet) = p(i^{-1}(\Gamma_1))^\bullet$. So let $\gamma \in Sen(\Sigma' \cap (\Sigma_1 \cup \Sigma_2))$. If $\gamma \in p(i^{-1}(\Gamma_1))^\bullet$, then $p(i^{-1}(\Gamma_1)) \models \gamma$, hence $j(p(i^{-1}(\Gamma_1))) \models j(\gamma)$, and furthermore, since $j(p(i^{-1}(\Gamma_1))) = k(i(i^{-1}(\Gamma_1))) \subseteq k(\Gamma_1)$, we get $k(\Gamma_1) \models j(\gamma)$, i.e., $\gamma \in j^{-1}(k(\Gamma_1)^\bullet)$. Conversely, assume $\gamma \in j^{-1}(k(\Gamma_1)^\bullet)$; then $k(\Gamma_1) \models j(\gamma)$; while this entailment holds within $\Sigma_1 \cup \Sigma_2$, it nevertheless holds within the "minimal" signature $\Sigma_1 \cup (\Sigma' \cap (\Sigma_1 \cup \Sigma_2)) = (\Sigma_1 \cup \Sigma') \cap (\Sigma_1 \cup \Sigma_2)$ too; by interpolation, we can find a set Γ of $\Sigma_1 \cap (\Sigma' \cap (\Sigma_1 \cup \Sigma_2))$-sentences, i.e., of $\Sigma' \cap \Sigma_1$-sentences, such that $\Gamma_1 \models i(\Gamma)$ and $p(\Gamma) \models \gamma$, hence, since Γ_1 is $\bullet$-closed, $\gamma \in p(\Gamma)^\bullet \subseteq p(i^{-1}(\Gamma_1))^\bullet$, and the proof is done.

Thus our interpolation results show that the restricted distributivity law holds in module algebra developed within many logical frameworks intermediate between full first-order logic and equational logic. Note that a property weaker than restricted distributivity, obtained by taking as further hypothesis that Σ' is included in both Σ_1 and Σ_2, was called *weak distributivity* in [14] and showed to hold regardless of interpolation; however, this extra hypothesis is rather restrictive because it asks that Σ_1 and Σ_2 interact with Σ in the same way.

[8] These operators are called *sum* and *information hiding* in [14].

Another application to formal specifications relies on the fact that interpolation entails a compositional behavior of the semantics of structured specifications, by ensuring that the two alternative semantics, the flat and the structured ones, coincide. This is crucial for keeping the semantics simple and amenable.

Let (Σ, Γ) be a theory and Σ' a subsignature of Σ. Sometimes one might prefer to *keep the structure* of, rather than flatten, the expression, $\Sigma' \Box (\Sigma, \Gamma)$ and consider Γ not as a theory, but as a presentation of a theory, $\Gamma^\bullet$ (this approach is taken for instance in [31, 6, 20]). Generally, the couple $(\Sigma' \hookrightarrow \Sigma, \Gamma))$ provides more information than $(\Sigma', \Gamma^\bullet \cap Sen(\Sigma'))$, for at least two reasons: 1) Γ might be finite, showing that $\Gamma^\bullet$, maybe unlike $\Gamma^\bullet \cap Sen(\Sigma')$, is finitely presented; 2) while the theory of all Σ'-reducts of (Σ, Γ) (i.e., all visible parts of the possible implementations of the theory) is indeed $\Gamma^\bullet \cap Sen(\Sigma')$, usually not any model of $\Gamma^\bullet \cap Sen(\Sigma')$ is a Σ-reduct of (Σ, Γ), unless a strong conservativity assumption is taken; hence the theory does not perfectly describe the intended semantics on classes of models. As mentioned, Σ' is to be regarded as the interface, or the visible signature, of the (hidden) theory (Σ, Γ).

Let us call, like in [20], a *module* the couple $(\Sigma' \hookrightarrow \Sigma, \Gamma)$. Assume now that one needs to import this module. A first step to take is to extend the visible signature, perhaps renaming some syntactic items to avoid undesired overlapping. This process is performed by the renaming operator $\star$, via a signature morphism $j : \Sigma' \to \Sigma''$. Subsequently to renaming, one will add further sentences to the extended module, but let us concentrate on renaming.

Thus what should $(\Sigma' \hookrightarrow \Sigma, \Gamma) \star j$ mean? If, for reasons mentioned above, one prefers to keep the modular structure, $(\Sigma' \hookrightarrow \Sigma, \Gamma) \star j$ should mean $(\Sigma' \hookrightarrow \Sigma, \Gamma) \star j$ and nothing else!, with natural semantics, consisting of all Σ''-models whose j-reducts have a Σ-expansion satisfying Γ; the Σ''-theory of such models is obviously $j(\Gamma^\bullet \cap Sen(\Sigma'))^\bullet$. On the other hand, the translated module itself might be regarded as a "simple" module, having Σ'' as the visible signature and importing the hidden structure of the module $(\Sigma' \hookrightarrow \Sigma, \Gamma)$; this is achieved by taking the pushout $(\Sigma'' \hookrightarrow \Sigma_0, j_0 : \Sigma \to \Sigma_0)$ of $(\Sigma' \hookrightarrow \Sigma, j : \Sigma' \to \Sigma'')$, yielding the new module $(\Sigma'' \hookrightarrow \Sigma_0, j_0(\Gamma))$.

There is the question whether these two approaches, the modular and the "flat" one, are equivalent. And indeed, thanks to interpolation they are so, at least w.r.t. the visible properties: $j(\Gamma^\bullet \cap Sen(\Sigma'))^\bullet = j_0(\Gamma)^\bullet \cap Sen(\Sigma'')$.

Again, this desirable semantical equivalence is shown by our results to hold for several first-order sublogics, qualifying them as suitable specification frameworks.

More precisely, lines 3,5,15 in the table preceding Theorem 2 show that the framework may be restricted to positive-, universal-, or [universal quantification of atom disjunction]- logics. Moreover, line 21 shows the same thing for [universal quantification of possibly infinite atom disjunction]-logic. According to these results, the renaming morphism j can be allowed to be injective *on sorts* in the case of positive logic and *any* morphism in the other three cases.

Note that lines 2,4,14,20 list results complementary to the above, and generalize those in [10]. These latter results relax the requirements not on the renaming morphism, but on the *hiding morphism* (allowing one to replace the inclusion $\Sigma' \hookrightarrow \Sigma$ with an arbitrary signature morphism).

Within a specification framework, one should not commit herself to a particular kind of first-order sub-logic, but rather use the available power of expression on a by-need basis, keeping flexible the border between expressive power and effective/efficient decision or computation. The issue of coexistence of different logical systems brings up a third application of our results.

Practice in formal specification has shown that the various logical systems that one would like to use should not be simply "swallowed" by a richer universal logic that encompasses them all, but rather integrated using logic translations. This methodology, which is the meta-logical counterpart of keeping structured (i.e., unflattened) the specifications themselves, is followed for instance in CafeObj [12, 13].

The underlying logical structure of this system can be formalized as a *Grothendieck institution* [9], which provides a means of building specifications inside the minimal needed logical system between the available ones. The framework is initially presented as an *indexed institution*, i.e., a family of logical systems (institutions) with translations (morphisms or comorphisms) between them, and than flattened by a Grothendieck construction.

The interpolation property was of course a natural test which had to be applied to this construction. Lifting interpolation from the component institutions to the Grothendieck institution was studied in [11]; a criterion is given there for lifting interpolation, consisting mainly of three conditions: (1) that the component institutions have interpolation (for some designated pushouts of signatures); (2) that the involved institution comorphisms have interpolation; (3) that each pullback in the index category yields an interpolating square of comorphisms. There is no space here for entering many details, but the interested reader is referred to [11].

We give just one example showing that, via the above conditions, some of our interpolation results can be used for putting together in a consistent way two very interesting logical systems: (finitary) first-order logic ($\mathcal{FO}$) and the logic of universally quantified possibly infinite conjunctions of atoms ($\forall\vee_\infty$). While the former is a well-established logic in formal specifications, the latter has the ability of expressing some important properties, not expressible in the former, such as *accessibility* of models, e.g., $(\forall x)(x = 0 \vee x = s(0) \vee x = s(s(0)) \vee \ldots)$ for natural numbers. If one combines these two logics, initiality conditions are also available, e.g., the above accessibility condition ("no junk") can be complemented with the "no confusion" statement $\neg \bigvee_{i,j\in\mathbb{N},i<j} s^i(0) = s^j(0)$. Since the two logical systems have the same signatures, condition (2) above is trivially satisfied. Moreover, our results stated in lines 1 and 21 of the table preceding Theorem 2 ensure condition (1) for some very wide class of signature pushouts. Finally, condition (3) is fulfilled by the result in line 22, which states that formulae from the two logics have interpolants *in their intersection logic*, that of universally quantified (finite) conjunctions of atoms.

6 Related work and concluding remarks

The idea of using axiomatizability properties for proving Craig interpolation first appeared, up to our knowledge, in [29] in the case of many-sorted equational logic. Then [28] generalized this to an arbitrary pullback of categories, by considering some Birkhoff-like operators on those categories, with results applicable to different versions of equational logic. An institution-independent relationship between Birkhoff-like axiomatizability and Craig interpolation was depicted in [10], using a concept of *Birkhoff institution*. If we disregard combination of logics and flatten to the least logic, the results in lines 2,4,14,20 of the table preceding Theorem 2 can be also found in [10].

Our Theorem 1 generalized the previous "semantic" results, bringing the technique of semantic interpolation, we might say, up to its limit. The merit of Theorem 1 is that, by analysing the clearly identified "semantic field" where an interpolant has a chance to be, provided general conditions under which a semantic interpolant has a syntactic counterpart (i.e., it is axiomatizable). But this theorem solved only half of the interpolation problem; concrete lifting and preserving conditions, as well as certain inclusions between operators, then had to be proved. Thus in this paper, we provided a general methodology for proving interpolation results, but also followed this methodology, by working out many concrete examples. The list of sub-first-order-logics that fit our framework is of course open for other suitably axiomatizable logics; and so are the possible combinations between these logics, which might guarantee interpolants even simpler than the types of formulae of *both* logics, as shown by some of our results.

Regarding our combined interpolation results, it is worth pointing out (and in fact the discussion at the end of Section 5 already hinted) that they are not overlapped with, but rather complementary to, the ones in [11] for Grothendieck institutions. There, some combined interpolation properties are previously assumed, in order to ensure interpolation in the resulted larger logical system.

An interesting fact to investigate would be to which extent can syntactically-obtained interpolation results "compete" with our semantic results. While it is true that the syntactic proofs are usually constructive, they do not seem to provide information on the type of the interpolant comparable to what we gave here. In particular, since the diverse Gentzen systems for first-order logic with equality have only partial cut elimination [17], an appeal to the non-equality version of the language, by adding appropriate axioms for equality in the theory, is needed; moreover, dealing with function symbols requires a further appeal to an encoding of functions as relations, again with the cost of adding some axioms. All these transformations make even some presumably very careful syntactic proofs rather indirect and obliterating, and sometimes place the interpolant way outside the given subtheory - this is the reason why an interpolation theorem for equational logic was not known until a separate, specific proof was given in [30]. Yet, comparing and paralleling (present or future) semantic and syntactic proofs seems fruitful for deepening our understanding of Craig interpolation, this extremely complex and resourceful, purely syntactic and yet surprisingly semantic, property of logical systems.

References

1. H. Andréka and I. Németi. A general axiomatizability theorem formulated in terms of cone-injective subcategories. In *Universal Algebra*, pages 13–36. North Holland, 1982. Colloquia Mathematica Societatis János Bolyai, 29.
2. J. Barwise and J. Feferman. *Model-Theoretic Logics.* Springer, 1985.
3. J. Bergstra, J. Heering, and P. Klint. Module algebra. *Journal of the ACM*, 37(2):335–372, 1990.
4. J. Bergstra, J. Heering, and P. Klint. Module algebra. *JACM*, 37(2):335–372, 1990.
5. T. Borzyszkowski. Generalized interpolation in CASL. *Inf. Process. Lett.*, 76(1-2):19–24, 2000.
6. T. Borzyszkowski. Logical systems for structured specifications. *Theor. Comput. Sci.*, 286(2):197–245, 2002.
7. C. C. Chang and H. J. Keisler. *Model Theory.* North Holland, Amsterdam, 1973.
8. W. Craig. Linear reasoning. A new form of the Herbrand-Gentzen Theorem. *Journal of Symbolic Logic*, 22:250–268, 1957.
9. R. Diaconescu. Grothedieck institutions. *Appl. Categorical Struct.*, 10(4):383–402, 2002.
10. R. Diaconescu. An institution-independent proof of Craig interpolation theorem. *Studia Logica*, 77(1):59–79, 2004.
11. R. Diaconescu. Interpolation in Grothendieck institutions. *Theor. Comput. Sci.*, 311:439–461, 2004.
12. R. Diaconescu and K. Futatsugi. *CafeOBJ Report.* World Scientific, 1998. AMAST Series in Computing, volume 6.
13. R. Diaconescu and K. Futatsugi. Logical foundations of CafeOBJ. *Theoretical Computer Science*, 285:289–318, 2002.
14. R. Diaconescu, J. Goguen, and P. Stefaneas. Logical support for modularization. In *Logical Environments*, pages 83–130. Cambridge, 1993.
15. R. Diaconescu, J. Goguen, and P. Stefaneas. Logical support for modularization. In G. Huet and G. Plotkin, editors, *Logical Environments*, pages 83–130. Cambridge, 1993.
16. T. Dimitrakos and T. Maibaum. On a generalized modularization theorem. *Information Processing Letters*, 74(1–2):65–71, 2000.
17. J. H. Gallier. *Logic for computer science. Foundations of automatic theorem proving.* Harper & Row, 1986.
18. J. Goguen. Types as theories. In *Topology and Category Theory in Computer Science*, pages 357–390. Oxford, 1991.
19. J. Goguen and R. Burstall. Institutions: Abstract model theory for specification and programming. *Journal of the ACM*, 39(1):95–146, January 1992.
20. J. Goguen and G. Roşu. Composing hidden information modules over inclusive institutions. In *From Object Orientation to Formal Methods: Dedicated to the memory of Ole-Johan Dahl*, volume 2635 of *LNCS*, pages 96–123. Springer, 2004.
21. D. Găină and A. Popescu. An institution-independent proof of Robinson consistency theorem. *Studia Logica*, to appear. Ask authors for current draft at popescu2@cs.uiuc.edu.
22. D. G. J. Bicarregui, T. Dimitrakos and T. Maibaum. Interpolation in practical formal development. *Logic Journal of the IGPL*, 9(1):231–243, 2001.
23. J. Meseguer. Membership algebra as a logical framework for equational specification. In *Proc. of WADT'97*, volume 1376 of *LNCS*, pages 18–61. Springer, 1998.
24. J. D. Monk. *Mathematical Logic.* Springer-Verlag, 1976.

25. G. Nelson and D. C. Oppen. Simplification by cooperating decision procedures. *ACM Trans. Program. Lang. Syst.*, 1(2):245–257, 1979.
26. I. Németi and I. Sain. Cone-implicational subcategories and some Birkhoff-type theorems. In *Universal Algebra*, pages 535–578. North Holland, 1982. Colloquia Mathematica Societatis János Bolyai, 29.
27. D. C. Oppen. Complexity, convexity and combinations of theories. *Theoretical Computer Science*, 12:291–302, 1980.
28. G. Roşu and J. Goguen. On equational Craig interpolation. *Journal of Universal Computer Science*, 6:194–200, 2000.
29. P. H. Rodenburg. A simple algebraic proof of the equational interpolation theorem. *Algebra Universalis*, 28:48–51, 1991.
30. P. H. Rodenburg and R. van Glabbeek. An interpolation theorem in equational logic. Technical Report CS-R8838, CWI, 1988.
31. D. Sannella and A. Tarlecki. Specifications in an arbitrary institution. *Information and Control*, 76:165–210, 1988.
32. A. Tarlecki. Bits and pieces of the theory of institutions. In *Proc. Summer Workshop on Category Theory and Computer Programming*, volume 240 of *LNCS*, pages 334–360. Springer, 1986.