mathematics

Article

Interpolation and Uniform Interpolation in Quantifier-Free Fragments of Combined First-Order Theories

Silvio Ghilardi [1] and **Alessandro Gianola** [2,*]

[1] Department of Mathematics, Universitá degli Studi di Milano, 20133 Milan, Italy; silvio.ghilardi@unimi.it
[2] Faculty of Computer Science, Free University of Bozen-Bolzano, 39100 Bolzano, Italy
* Correspondence: gianola@inf.unibz.it

Abstract: In this survey, we report our recent work concerning combination results for interpolation and uniform interpolation in the context of quantifier-free fragments of first-order theories. We stress model-theoretic and algebraic aspects connecting this topic with amalgamation, strong amalgamation, and model-completeness. We give sufficient (and, in relevant situations, also necessary) conditions for the transfer of the quantifier-free interpolation property to combined first-order theories; we also investigate the non-disjoint signature case under the assumption that the shared theory is universal Horn. For convex, strong-amalgamating, stably infinite theories over disjoint signatures, we also provide a modular transfer result for the existence of uniform interpolants. Model completions play a key role in the whole paper: They enter into transfer results in the non-disjoint signature case and also represent a semantic counterpart of uniform interpolants.

Keywords: interpolation; combined interpolation; uniform interpolation; satisfiability modulo theories

check for updates

Citation: Ghilardi, S.; Gianola, A. Interpolation and Uniform Interpolation in Quantifier-Free Fragments of Combined First-Order Theories. *Mathematics* **2022**, *10*, 461. https://doi.org/10.3390/math10030461

Academic Editors: Cristina Sernadas and João Rasga

Received: 1 December 2021
Accepted: 13 January 2022
Published: 31 January 2022

1. Introduction

Craig's interpolation theorem [1] is a classical well-known result in first-order logic; it says that whenever an implication

$$\phi \to \psi \tag{1}$$

is valid, then there exists a formula θ such that the implications

$$\phi \to \theta \text{ and } \theta \to \psi \tag{2}$$

are valid too, and the formula θ contains at most the symbols occurring both in ϕ and in ψ. This theorem has been largely investigated both in propositional and predicate logics; a renewed interest in it has come from recent applications in verification [2–4]. The reason for why interpolation became important in verification is because it helps to discover, in a completely *automatic* way, new predicates that might contribute to the construction of invariants. In fact, many model-checking problems are of an *infinite state*, which means that the language needed in order to build, e.g., safety invariants, is quite rich and is far from requiring only finitely many formulae up to logical equivalence. Popular methods for synthesizing invariants analyze spurious error traces. Suppose that the system to be verified is specified via a triple $\langle \underline{x}, \iota(\underline{x}), \tau(\underline{x}, \underline{x}') \rangle$ given by a tuple of variables $\underline{x}$, a formula $\iota(\underline{x})$ describing initial states, and a formula $\tau(\underline{x}, \underline{x}')$ describing state evolutions; suppose also that we are given a further formula $v(\underline{x})$ describing undesired 'error' states. Then, the system under examination cannot reach an error configuration in n steps iff the formula

$$\iota(\underline{x}_0) \wedge \tau(\underline{x}_0, \underline{x}_1) \wedge \cdots \wedge \tau(\underline{x}_{n-1}, \underline{x}_n) \wedge v(\underline{x}_n)$$

is not satisfiable (in the models of a suitable theory T). From the unsatisfiability proof, taking an interpolant, say, at the i-th iteration, one can produce a formula $\theta(\underline{x})$ such that, modulo T, we have that the implications

$$\iota(\underline{x}_0) \wedge \bigwedge_{j=1}^{i} \tau(\underline{x}_{j-1}, \underline{x}_j) \to \theta(\underline{x}_i) \quad \text{and} \quad \theta(\underline{x}_i) \wedge \bigwedge_{j=i+1}^{n} \tau(\underline{x}_{j-1}, \underline{x}_j) \wedge \upsilon(\underline{x}_n) \to \bot \tag{3}$$

are both T-valid (i.e., true in all models of T). The formula θ (and the literals it contains) can contribute to the refinement of the current candidate safety invariant. This fact is exploited in different ways during invariant search; it can also be combined with orthogonal techniques in existing implementations, as witnessed by a rich literature; see, e.g., [4–10], among many other contributions.

One major problem encountered during the above applications concerns the *shape* of the interpolant. Usually, one considers implications such as (1), which are valid in all models of a given first-order theory T; but in general, there is no guarantee that if ϕ, ψ are both quantifier-free, then there is an interpolant θ such that the implications (2) are T-valid and such that θ is also quantifier-free. This is crucial because, very often, first-order theories commonly used in verification have a decidable quantifier-free fragment, but are undecidable outside that fragment (this is the case, for instance, of the McCarthy theory of arrays; see Section 3 below); even if general first-order satisfiability remains decidable, the computational cost of a satisfiability test may increase considerably when moving from the quantifier-free fragment to arbitrary first-order formulae (this is the case, for instance, of Presburger arithmetic). This is why some considerable effort has been put into designing theory-specific interpolation algorithms operating at a quantifier-free level [11–17] and in identifying suitable variants of theories axiomatizing common datatypes enjoying *quantifier-free interpolation* [18–20].

Still, knowing that an isolated theory by itself has quantifier-free interpolation might not be sufficient for applications; in common benchmarks, it happens that arrays, sets, lists, etc. are always arrays, sets, lists *of* something (booleans, integers, reals, etc.), so that one must be sure that quantifier-free interpolation *transfers* to combined theories. This will be the main subject of the first part of the present survey paper.

In the second part of the paper, we consider a strong form of Craig interpolation, namely, *uniform* interpolation. We recall here what uniform interpolants are in the context of the quantifier-free fragment of a first-order theory T. We use notations such as $\psi(\underline{x})$ to say that at most the variables from the tuple $\underline{x}$ occur freely in ψ. Given a quantifier-free formula $\phi(\underline{x}, \underline{y})$, a *uniform interpolant* of ϕ (w.r.t. $\underline{y}$) is a quantifier-free formula $\theta(\underline{x})$ satisfying the following two properties:

- $\phi(\underline{x}, \underline{y}) \to \theta(\underline{x})$ is T-valid;
- For any further quantifier-free formula $\psi(\underline{x}, \underline{z})$ such that $\phi(\underline{x}, \underline{y}) \to \psi(\underline{x}, \underline{z})$ is T-valid, we have that the implication $\theta(\underline{x}) \to \psi(\underline{x}, \underline{z})$ is T-valid, too.

Whenever uniform interpolants exist, one can compute an interpolant for an entailment such as $\phi(\underline{x}, \underline{y}) \to \psi(\underline{x}, \underline{z})$ in a way that is *independent* of ψ. Uniform interpolants have been widely studied in the context of non-classical propositional logics (a non-exhaustive list includes [21–28]). In the last decade, the automated reasoning community has also developed an increasing interest in uniform interpolants, this time for quantifier-free fragments of first-order theories [29,30]; in this literature, uniform interpolants are often called 'covers', but the definitions of uniform interpolants and of covers are equivalent. In these contributions, examples of computations of uniform interpolants were supplied, and some algorithms were also sketched. The first formal *proofs* about the existence of uniform interpolants in $\mathcal{EUF}$ (the theory of pure equality in an arbitrary signature) were however published only in [31,32]. The usefulness of uniform interpolants in model checking was already stressed in [30] and further motivated by our recent line of research on the verification of data-aware processes [31,33–35]. In such applications, combination problems obviously arise, so in Section 5 below, we investigate transfer problems for uniform quantifier-free interpolation.

Structure of the Paper

Section 2 settles on notations and basic definitions. Section 3 investigates quantifier-free interpolation for combined theories in disjoint signatures and the semantic counterparts related to quantifier-free interpolation: amalgamation, strong amalgamation, and definability properties. Section 4 extends this analysis to the case of non-disjoint signature theories and shows interesting applications to modal logic. Section 5 introduces uniform interpolants, discusses their existence in $\mathcal{EUF}$ and shows how to transfer them to combined convex theories. Section 6 concludes.

The paper is conceived as a survey paper, principally addressed to a mathematical audience; however, it should be taken into account that many motivations and examples that suggested the results included in the paper come from the software verification area; hence, we give at least some sketches of algorithmic aspects. Proofs are omitted (sometimes in favor of intuitive justifications); however, they can all be found in the original papers. More precisely, proofs of the results from Section 3 are in [36], proofs of the results from Section 4 are in [37], and proofs of the results from Section 5 are in [31,32,38–40].

2. Preliminaries

We assume that the reader is familiar with the basic notions concerning first-order logic; this includes syntactic notions such as signature, variable, term, atom, literal, formula, and sentence and semantic notions such as structure, substructure, truth, satisfiability, and validity. The equality symbol "=" is considered a logical symbol and, hence, is included in all signatures considered below; to exclude limit cases, we always assume that our signatures always contain at least one individual constant symbol. When we use notations such as $E(\underline{x})$, we mean that the expression (term, literal, formula, etc.) E contains free variables only from the tuple $\underline{x}$. Concerning 'tuples', we make an important convention: When we speak of a 'tuple of variables', the tuple is meant to represent an arity; hence, it is assumed not to contain repetitions. The same convention does not apply when we speak of a 'tuple of terms', which might consequently contain repetitions. These conventions are useful for substitutions; we use them when denoting with $\phi(\underline{t}/\underline{x})$ the formula obtained from $\phi(\underline{x})$ by the simultaneous replacement of the 'tuple of variables' $\underline{x}$ with the 'tuple of terms' $\underline{t}$. For similar reasons, whenever we use a notation such as $E(\underline{x}, \underline{y})$, we assume not only that the tuples $\underline{x}$ and $\underline{y}$ are made of pairwise distinct elements, but also that $\underline{x}$ and $\underline{y}$ are disjoint as sets. A formula is *universal* (*existential*) iff it is obtained from a quantifier-free formula by prefixing it with a string of universal (or existential) quantifiers. A formula is *ground* iff it does not contain occurrences of variables (neither free nor bound).

From the semantic side, we refer to standard model-theoretic terminology [41] for basic notions such as structures, embeddings, diagrams, etc. Σ-structures are indicated with calligraphic letters $\mathcal{A}, \mathcal{B}, \dots, \mathcal{M}, \mathcal{N}, \dots$; the support of a Σ-structure $\mathcal{A}$ is indicated with $|\mathcal{A}|$. A *theory* T in a signature Σ is a set of Σ-sentences; the *models* of T are those Σ-structures in which all the sentences in T are true. A Σ-formula ϕ is T-*satisfiable* (or T-consistent) if there exists a model $\mathcal{M}$ of T such that ϕ is true in $\mathcal{M}$ under a suitable assignment $\mathtt{a}$ to the free variables of ϕ (in symbols, $(\mathcal{M}, \mathtt{a}) \models \phi$); it is T-*valid* (in symbols, $T \vdash \varphi$) if its negation is not T-satisfiable. A theory T is *universal* iff all sentences in T are universal. A theory T admits *quantifier-elimination* iff, for every formula $\phi(\underline{x})$, there is a quantifier-free formula $\phi'(\underline{x})$ such that $T \vdash \phi \leftrightarrow \phi'$ (since we work in a computational logic context, we consider part of the definition of a theory enjoying quantifier elimination the fact that such ϕ' is effectively computable from ϕ). A formula φ_1 T-*entails* a formula φ_2 if the implication $\varphi_1 \to \varphi_2$ is T-*valid* (in symbols, $\varphi_1 \vdash_T \varphi_2$, or simply $\varphi_1 \vdash \varphi_2$ when T is clear from the context). If Γ is a set of formulæ and ϕ is a formula, the notation $\Gamma \vdash_T \phi$ means that there are $\gamma_1, \dots, \gamma_n \in \Gamma$ such that $\gamma_1 \wedge \dots \wedge \gamma_n \vdash_T \phi$. The *satisfiability modulo the theory* T (SMT(T)) *problem* amounts to establishing the T-satisfiability of quantifier-free Σ-formulæ. Some theories have special standard names in the SMT-literature (some of these names will be recalled during the paper). For pure equality theory, our conventions are as follows. We shall call $\mathcal{EUF}(\Sigma)$ the pure equality theory in the signature Σ; we may also use just $\mathcal{EUF}$

instead of $\mathcal{EUF}(\Sigma)$ in case there is no need to specify the signature Σ; however, in that case, Σ is assumed to be *proper*, i.e., it must contain (besides free constants) at least a predicate or a function symbol different from equality.

2.1. Combinations of Theories

Stable infiniteness is a semantic ingredient often occurring in combination results; the requirement is rather mild, and most theories used in verification, such as theories axiomatizing fragments of arithmetics as well as common datatypes, satisfy it (but there are also notable exceptions, such as bitvector theories). The formal definition is as follows. A theory T is *stably infinite* iff every T-satisfiable quantifier-free formula (from the signature of T) is satisfiable in an infinite model of T. By compactness, it is immediate to show that T is stably infinite iff every model of T embeds into an infinite one.

Let T_i be a stably infinite theory over the signature Σ_i such that the $SMT(T_i)$ problem is decidable for $i = 1, 2$ and Σ_1 and Σ_2 are disjoint (i.e., the only shared symbol is equality). Under these assumptions, the *Nelson–Oppen combination method* [42,43] tells us that the SMT problem for the combination $T_1 \cup T_2$ of the theories T_1 and T_2 (i.e., the union of their axioms) is decidable. In general, however, the combined SMT problem $T_1 \cup T_2$ may become undecidable, even when the $SMT(T_1)$, $SMT(T_2)$ problems are decidable and the signatures are disjoint [44]; on the other hand, stable infiniteness is a sufficient but not necessary condition for the decidability transfer of SMT problems to a disjoint combination (for a survey on different sufficient criteria, see [45], and for recent developments, also see [46]).

2.2. Interpolation Properties

In this paper, we are interested in specializing the Craig interpolation property to quantifier-free fragments of first-order theories. We give two definitions: The first one is more restricted, and the second one is more liberal.

Definition 1. *[Plain quantifier-free interpolation] A theory T* admits (plain) quantifier-free interpolation *(or, equivalently,* has quantifier-free interpolants*) iff, for every pair of quantifier-free formulae $\phi(\underline{x}, \underline{y})$, $\psi(\underline{y}, \underline{z})$ such that $\psi \vdash_T \phi$, there exists a quantifier-free formula $\theta(\underline{y})$, called an* interpolant, *such that: (i) $\psi \vdash_T \theta$, (ii) $\theta \vdash_T \phi$. (Notice that only the variables $\underline{y}$ occurring in both ψ and ϕ can occur in θ.)*

The following extension of the above definition is considered more natural (and also more useful in verification applications):

Definition 2. *[General quantifier-free interpolation] Let T be a theory in a signature Σ; we say that T has the* general quantifier-free interpolation property *iff, for every signature Σ' (disjoint from Σ) and for every pair of ground $\Sigma \cup \Sigma'$-formulæ ϕ, ψ such that $\psi \vdash_T \phi$, there exists a ground formula θ, such that: (i) $\psi \vdash_T \theta$, (ii) $\theta \vdash_T \phi$, and (iii) all predicates, constants, and function symbols from Σ' occurring in θ also occur both in ϕ and ψ.*

Since free variables can be replaced by free constants without affecting entailment relations, it should be clear that the general quantifier-free interpolation property (Definition 2) implies the plain quantifier-free interpolation property (Definition 1).

2.3. Amalgamation Properties

When stating amalgamability and strong amalgamability properties (see [47] for a survey), people usually limit themselves to universal theories; actually, most theories we have in mind for several applications are universal; however, to some extent, we also want to handle general first-order theories in the paper. In order to do that, it is important to observe that a substructure of a model of a non-universal theory need not be a model of the theory. Thus, in our definitions, we must take into account substructures that are not necessarily submodels. This leads to the notions below, which we call 'sub-amalgamability' and 'strong sub-amalgamability':

Definition 3. *Let T be a theory; we call a T-fork a triple $(\mathcal{M}_1, \mathcal{M}_2, \mathcal{A})$, where $\mathcal{M}_1, \mathcal{M}_2$ are models of T and $\mathcal{A}$ is their shared substructure. (By this, we mean that $\mathcal{A}$ is a substructure of both $\mathcal{M}_1$and$\mathcal{M}_2$ and that $|\mathcal{A}| = |\mathcal{M}_1| \cap |\mathcal{M}_2|$.) A T-*amalgam *of such a fork is a triple $(\mathcal{M}, \mu_1, \mu_2)$, where $\mathcal{M}$ is a T-model and $\mu_1 : \mathcal{M}_1 \longrightarrow \mathcal{M}$, $\mu_2 : \mathcal{M}_2 \longrightarrow \mathcal{M}$ are embeddings whose restrictions to the support of $\mathcal{A}$ coincide. A theory T has the* sub-amalgamation property *iff every T-fork has a T-amalgam.*

$$\begin{tikzcd}
\mathcal{M}_1 \arrow[r, "\mu_1"] & \mathcal{M} \\
\mathcal{A} \arrow[u] \arrow[r] & \mathcal{M}_2 \arrow[u, "\mu_2"']
\end{tikzcd}$$

A theory T has the strong sub-amalgamation property *iff every T-fork $(\mathcal{M}_1, \mathcal{M}_2, \mathcal{A})$ has a T-amalgam $(\mathcal{M}, \mu_1, \mu_2)$ satisfying the following additional condition: If for some $m_1 \in |\mathcal{M}_1|, m_2 \in |\mathcal{M}_2|$, we have $\mu_1(m_1) = \mu_2(m_2)$, then there exists an element a in $|\mathcal{A}|$ such that $m_1 = a = m_2$.*

When the theory T is universal, we may speak of 'amalgamation' and 'strong amalgamantion' properties instead of 'sub-amalgamation' and 'strong sub-amalgamation' properties, respectively.

3. Strong Amalgamation and Combined Interpolation

The results presented in this section concern the relationships between syntactical notions, such as forms of interpolation, and their corresponding semantic counterparts, such as variants of amalgamation, and they are based on [36]. An old result due to Bacsich [48] connects quantifier-free interpolation and amalgamation for the case of universal theories; the result can be easily extended to arbitrary first-order theories replacing amalgamation with sub-amalgamation [36].

Theorem 1 ([36,48])**.** *A theory T has the sub-amalgamation property iff it admits quantifier-free interpolants.*

The above theorem can be used to find examples and counterexamples. For instance, it is easily seen that $\mathcal{EUF}$ (which is trivially universal) has amalgamation, and hence, it has quantifier-free interpolants; in fact, $\mathcal{EUF}$ also has the strong amalgamation property mentioned above. A simple example of a universal theory that does *not* enjoy amalgamation is the theory of a binary relation that is a partial function.

Less trivial examples and counterexamples are given by the variants of McCarthy's theory of arrays [49]. We consider three variants of this theory. The first variant is $\mathcal{AR}_{\text{ext}}$, which is the theory of arrays with extensionality. The signature of $\mathcal{AR}_{\text{ext}}$ contains the sort symbols `ARRAY`, `ELEM`, and `INDEX` and the function symbols $rd : \texttt{ARRAY} \times \texttt{INDEX} \longrightarrow \texttt{ELEM}$ and $wr : \texttt{ARRAY} \times \texttt{INDEX} \times \texttt{ELEM} \longrightarrow \texttt{ARRAY}$. (notice that Theorem 1, and in general all results in this paper, extends to many-sorted signatures). The set of axioms of $\mathcal{AR}_{\text{ext}}$ consists of the following three sentences:

$$\forall y, i, j, e.\ i \neq j \to rd(wr(y, i, e), j) = rd(y, j), \tag{4}$$

$$\forall y, i, e.\ rd(wr(y, i, e), i) = e, \tag{5}$$

$$\forall x, y.\ x \neq y \to (\exists i.\ rd(x, i) \neq rd(y, i)) \tag{6}$$

Now, $\mathcal{AR}_{\text{ext}}$ enjoys the amalgamation property in the sense that, given two models $\mathcal{M}_1$ and $\mathcal{M}_2$ of $\mathcal{AR}_{\text{ext}}$ sharing a substructure $\mathcal{M}_0$ *that is also a model of $\mathcal{AR}_{\text{ext}}$*, there is a model $\mathcal{M}$ of $\mathcal{AR}_{\text{ext}}$ endowed with embeddings from $\mathcal{M}_1, \mathcal{M}_2$ agreeing on the support of $\mathcal{M}_0$. However, $\mathcal{AR}_{\text{ext}}$ is *not* universal, so this is not sufficient to guarantee quantifier-free interpolation. In fact, $\mathcal{AR}_{\text{ext}}$ is not sub-amalgamable, and quantifier-free interpolation fails for it, as shown by the following valid implication whose interpolants require a quantifier (the counterexample is due to R. Jhala and is reported in [50]):

$$a = wr(b, i, e) \to \neg(j_1 \neq j_2 \wedge rd(a, j_1) \neq rd(b, j_1) \wedge rd(a, j_2) \neq rd(b, j_2)) \ . \tag{7}$$

The theory $\mathcal{AX}_{\mathtt{diff}}$ is obtained from $\mathcal{AR}_{\mathrm{ext}}$ by skolemizing the extensionality axiom (6); hence, its language has an extra binary function $\mathtt{diff} : \mathtt{ARRAY} \times \mathtt{ARRAY} \longrightarrow \mathtt{INDEX}$ and the following additional axiom:

$$\forall x, y. \qquad x \neq y \to rd(x, \mathtt{diff}(x, y)) \neq rd(y, \mathtt{diff}(x, y)) \ . \tag{8}$$

which replaces (6). This theory is universal and (strongly) amalgamable [18,19]. This means that quantifier-free interpolants exist; for example, an interpolant of the two formulæ (8) can be written without quantifiers in this theory as

$$a = wr(b, \mathtt{diff}(a, b), rd(b, \mathtt{diff}(a, b))) \ .$$

The third variant of the array theory we want to mention is the theory $\mathcal{AX}_{maxdiff}$, where the axiom (8) is strengthened so that $\mathtt{diff}(a, b)$ returns the *biggest* index where a, b differ. This requires adding at least a symbol for a total ordering relation on the sort INDEX (we leave the reader to consult [20] for details). Under suitable mild hypotheses, it is possible to prove that the universal theory $\mathcal{AX}_{maxdiff}$ also has amalgamation and, hence, quantifier-free interpolation (but the proof is surprisingly much more delicate [20]).

Amalgamation and sub-amalgamation are not modular properties in the sense that they can get lost when taking union of theories, even under disjoint signatures. However, *strong* amalgamation, under stable infiniteness, is modular [36].

Theorem 2 ([36])**.** *Let T_1 and T_2 be two stably infinite theories over disjoint signatures Σ_1 and Σ_2. If both T_1 and T_2 have the strong sub-amalgamation property, then so does $T_1 \cup T_2$. Thus, in view of Theorem 1, $T_1 \cup T_2$ has quantifier-free interpolants.*

Actually, strong sub-amalgamation is a necessary condition for the transfer of the quantifier-free interpolation property in the sense that is precisely stated in the following result.

Theorem 3 ([36])**.** *Let T be a theory admitting quantifier-free interpolation and let Σ be a proper signature disjoint from the signature of T. Then, $T \cup \mathcal{EUF}(\Sigma)$ has quantifier-free interpolation iff T has the strong sub-amalgamation property.*

The intuitive reason for why the above theorem holds is the following. Recall that since Σ is proper, it must contain at least a function or a predicate symbol different from equality. Suppose, e.g., it contains a unary predicate P; take models $\mathcal{M}_1, \mathcal{M}_2$ of T that cannot be strongly amalgamated over their common substructure. Then, expand them to Σ-structures in such a way that P holds precisely for the elements of the support of $\mathcal{M}_1$ that are not from the support of $\mathcal{M}_2$; clearly, sub-amalgamation fails for these expanded models; hence, $T \cup \mathcal{EUF}(\Sigma)$ lacks quantifier-free interpolation.

Strong amalgamation also characterizes the general quantifier-free interpolation property.

Theorem 4 ([36])**.** *A theory T has the general quantifier-free interpolation property iff T has the strong sub-amalgamation property.*

3.1. Strong Amalgamation: A Syntactic Characterization

Strong amalgamation needs an 'operational' characterization to be useful when designing concrete combined interpolation algorithms. We reformulate strong amalgamation via a syntactic property (to be called the equality-interpolating property); this syntactic property, roughly speaking, says that disjunctions of variables' equalities can be entailed, modulo T, by quantifier-free formulae only in the case that such equalities are mediated by explicitly defining terms.

Given two finite tuples $\underline{t} \equiv t_1, \ldots, t_n$ and $\underline{v} \equiv v_1, \ldots, v_m$ of terms,

$$\text{the notation } \underline{t} \cap \underline{v} \neq \emptyset \text{ stands for the formula } \bigvee_{i=1}^{n} \bigvee_{j=1}^{m} (t_i = v_j).$$

We use $\underline{t}_1\underline{t}_2$ to denote the juxtaposition of the two tuples $\underline{t}_1$ and $\underline{t}_2$ of terms. So, for example, $\underline{t}_1\underline{t}_2 \cap \underline{v} \neq \emptyset$ is equivalent to $(\underline{t}_1 \cap \underline{v} \neq \emptyset) \vee (\underline{t}_2 \cap \underline{v} \neq \emptyset)$. Next Definition is taken from [36] [Definition 4.1 and Theorem 4.2(iii)]:

Definition 4. *A theory T is* equality interpolating *iff it has the quantifier-free interpolation property and satisfies the following condition:*

- *for every triple $\underline{x}, \underline{y}_1, \underline{y}_2$ of tuples of variables and for every pair of quantifier-free formulae $\delta_1(\underline{x}, \underline{y}_1), \delta_2(\underline{x}, \underline{y}_2)$ such that*

$$\delta_1(\underline{x}, \underline{y}_1) \wedge \delta_2(\underline{x}, \underline{y}_2) \vdash_T \underline{y}_1 \cap \underline{y}_2 \neq \emptyset \tag{9}$$

  there exists a tuple $\underline{v}(\underline{x})$ of terms such that

$$\delta_1(\underline{x}, \underline{y}_1) \wedge \delta_2(\underline{x}, \underline{y}_2) \vdash_T \underline{y}_1\underline{y}_2 \cap \underline{v} \neq \emptyset\,. \tag{10}$$

The following theorem states the syntactic counterpart of the strong amalgamation property.

Theorem 5 ([36])**.** *Given a theory T with quantifier-free interpolation, the following conditions are equivalent:*

(i) *T is strongly sub-amalgamable;*
(ii) *T is equality interpolating.*

If a theory T has quantifier elimination, then it obviously also has quantifier-free interpolants, and hence, it is sub-amalgamable. However, quantifier elimination is not sufficient to get strong sub-amalgamation (see below for counterexamples). Nevertheless, if the theory is also universal, then quantifier elimination is sufficient.

Theorem 6 ([36])**.** *A universal theory admitting quantifier elimination is equality interpolating.*

Proof. (Sketch) In order to prove this theorem, one needs to preliminarily show a *testing-point quantifier elimination lemma*; such a lemma says that if T is universal and has quantifier elimination, then for every quantifier-free formula $\phi(\underline{x}, \underline{y})$, there exists a tuple $\underline{t}_1(\underline{x}), \ldots, \underline{t}_n(\underline{x})$ of tuples of terms such that

$$T \vdash \exists \underline{y}\, \phi(\underline{x}, \underline{y}) \leftrightarrow \bigvee_{i=1}^{n} \phi(\underline{x}, \underline{t}_i(\underline{x}))\ \ . \tag{11}$$

Taking this preliminary result for granted, we formally prove that a universal and quantifier eliminable theory T satisfies the implication (9) ⇒ (10). Suppose that (9) holds; by the testing-point quantifier elimination lemma, there exist tuples of terms $\underline{t}_1(\underline{x}), \ldots, \underline{t}_k(\underline{x})$ such that

$$\exists \underline{y}_2\, \delta_2(\underline{x}, \underline{y}_2) \leftrightarrow \bigvee_{j=1}^{k} \delta_2(\underline{x}, \underline{t}_j(\underline{x})) \tag{12}$$

is T-valid. For every $j = 1, \ldots, k$, if we replace $\underline{y}_2$ with $\underline{t}_j$ in (9), we get

$$\delta_1(\underline{x}, \underline{y}_1) \wedge \delta_2(\underline{x}, \underline{t}_j) \vdash_T \underline{y}_1 \cap \underline{t}_j \neq \emptyset$$

and, hence,

$$\delta_1(\underline{x}, \underline{y}_1) \wedge \bigvee_{j=1}^{k} \delta_2(\underline{x}, \underline{t}_j) \vdash_T \bigvee_{j=1}^{k} (\underline{y}_1 \cap \underline{t}_j \neq \emptyset) \ .$$

Taking into account (12) and letting $\underline{v}$ be the tuple $\underline{t}_1 \cdots \underline{t}_k$ obtained by juxtaposition, we get

$$\delta_1(\underline{x}, \underline{y}_1) \wedge \exists \underline{y}_2 \delta_2(\underline{x}, \underline{y}_2) \vdash_T \underline{y}_1 \cap \underline{v} \neq \emptyset \ .$$

Removing the existential quantifier in the antecedent of the implication, we obtain

$$\delta_1(\underline{x}, \underline{y}_1) \wedge \delta_2(\underline{x}, \underline{y}_2) \vdash_T \underline{y}_1 \cap \underline{v} \neq \emptyset$$

and, a fortiori, (10), as desired. □

Theorem 6 immediately yields a bunch of strongly amalgamating theories.

Example 1. *The theory $\mathcal{RDS}$ of* recursive data structures *[51] requires a signature comprising two unary function symbols 'car' and 'cdr' and a binary function symbol 'cons'; the axioms of $\mathcal{RDS}$ are the following:*

$$\forall x, y. car(cons(x, y)) = x, \tag{13}$$
$$\forall x, y. cdr(cons(x, y)) = y, \tag{14}$$
$$\forall x, y. cons(car(x), cdr(x)) = x, \tag{15}$$
$$\forall x. x \neq t(x), \tag{16}$$

where t is a term obtained by finitely many applications of car and cdr to the variable x (e.g., axioms (16) include $\forall x. car(x) \neq x$, $\forall x. cdr(cdr(x)) \neq x$, $\forall x. cdr(car(x)) \neq x$, and so on). Clearly, $\mathcal{RDS}$ is universal; the fact that it admits elimination of quantifiers has been known since an old work by Mal'cev [52].

Example 2. *The theory $\mathcal{IDL}$ of* integer difference logic *requires a signature comprising the constant symbol '0', the unary function symbols 'succ' and 'pred', and the binary predicate symbol '$<$'; it is axiomatized by adding to the irreflexivity, transitivity, and linearity axioms for $<$ the following set of sentences:*

$$\forall x. succ(pred(x)) = x, \qquad \forall x. pred(succ(x)) = x,$$
$$\forall x, y. x < succ(y) \leftrightarrow (x < y \vee x = y), \qquad \forall x, y. pred(x) < y \leftrightarrow (x < y \vee x = y).$$

$\mathcal{IDL}$ is universal, and the fact that admits elimination of quantifiers can be shown by slightly modifying the procedure for the similar theory of natural numbers with successor and ordering in [53]. Notice that the atoms of $\mathcal{IDL}$ are equivalent to formulae of the form $i \bowtie f^n(j)$, where

(a) *$n \in \mathbb{Z}$ and $\bowtie \in \{=, <\}$;*
(b) *i, j are variables or the constant 0;*
(c) *$f^0(j)$ is j, $f^k(j)$ abbreviates $succ(succ^{k-1}(j))$ when $k > 0$ or $pred(pred^{k-1}(j))$ when $k < 0$.*

Usually, $i \bowtie f^n(j)$ is written as $i - j \bowtie n$ or as $i \bowtie j + n$; hence, the name of "integer difference logic."

Example 3. *The theory $\mathcal{UTVPI}$ is a fragment of linear arithmetic over the integers that is slightly more expressive than $\mathcal{IDL}$. It can be defined as the theory whose axioms are the sentences that are true in $\mathbb{Z}$ in the signature comprising the constant 0, the unary function symbols pred, succ, and $-$, and the binary predicate symbol $<$. It can be shown that $\mathcal{UTVPI}$ has a set of quantifier-eliminating universal axioms [36]; thus, $\mathcal{UTVPI}$ is equality interpolating.*

Example 4. *Linear Arithmetic over the Reals can be axiomatized as the theory of totally ordered divisible abelian groups [41]. It has quantifier elimination (e.g., via the Fourier–Motzkin procedure), but it is easily seen that it is not strongly sub-amalgamable (just consider two copies of the reals* *sharing the integers as a common substructure). However, if one includes* multiplication by rational coefficients *in the signature of the theory, one gets a universal set of axioms enjoying quantifier elimination, thus gaining strong amalgamation and the equality-interpolating property.*

Example 5. *The situation is somewhat similar for integer linear arithmetics. The theory of the integers under addition, 0, successor, and ordering does not have quantifier elimination; if we add infinitely many unary predicates for equivalence modulo n, we get Presburger arithmetics that enjoy quantifier elimination. However, this is not yet sufficient for the equality-interpolating property; for that, we must add infinitely many unary function symbols for integer division by n, varying n (see [36] or [54] for details).*

3.2. The Case of Convex Theories

A first-order theory T is said to be *convex* iff, for every conjunction of literals δ if

$$\delta \vdash_T \bigvee_{i=1}^{n} x_i = y_i,$$

then there is $i = 1, \ldots, n$ such that

$$\delta \vdash_T x_i = y_i \, .$$

Among convex theories, we have universal Horn theories (see Section 4 below); another remarkable example of a convex theory is linear real arithmetic (here is where the name 'convex' comes from: It comes from the fact that the convexity of linear real arithmetic follows from the geometrical fact that if a convex set is contained in a union of hyperplanes, then it is contained in one of them). On the other hand, integer linear arithmetic (Example 5) and integer difference logic (Example 2) are non-convex theories.

In convex theories, one can formulate the equality-interpolating condition in some interesting simpler ways. A formula is said to be *primitive* iff it is obtained by prefixing some existential quantifiers to a conjunction of literals.

Proposition 1 ([36]). *The following conditions are equivalent for a convex theory T with quantifier-free interpolation:*

(i) *T is equality interpolating;*

(ii) *For every pair y_1, y_2 of variables and for every pair of* conjunctions of literals $\delta_1(\underline{x}, \underline{z}_1, y_1)$, $\delta_2(\underline{x}, \underline{z}_2, y_2)$ *such that*

$$\delta_1(\underline{x}, \underline{z}_1, y_1) \wedge \delta_2(\underline{x}, \underline{z}_2, y_2) \vdash_T y_1 = y_2 \tag{17}$$

there exists a term $v(\underline{x})$ such that

$$\delta_1(\underline{x}, \underline{z}_1, y_1) \wedge \delta_2(\underline{x}, \underline{z}_2, y_2) \vdash_T y_1 = v(\underline{x}) \wedge y_2 = v(\underline{x}). \tag{18}$$

(iii) *For every tuple of variables $\underline{x}$, for every further variable y, and for every primitive formula $\delta(\underline{x}, y)$ such that*

$$\delta(\underline{x}, y') \wedge \delta(\underline{x}, y'') \vdash_T y' = y''$$

there is a term $v(\underline{x})$ such that

$$\delta(\underline{x}, y) \vdash_T y = v(\underline{x}) \, .$$

Condition (ii) is due to Yorsh and Musuvathi; in fact, in [55], they proved the combination Theorem 2 for the restricted case of convex theories using precisely condition (ii) instead of the semantic notion of strong sub-amalgamation. Condition (iii) is the Beth definability property formulated for the primitive fragment of the language—we shall call it the *primitive Beth definability property* (modulo T). We shall make use of this property in

Section 5.2 when we discuss an algorithm for computing combined uniform interpolants in the convex case.

3.3. Sketch of the Combined Interpolation Algorithm

Theorem 2 shows that the union of two stably infinite signature-disjoint strongly amalgamable theories has the quantifier-free interpolation property. However, it does not show how to compute quantifier-free interpolants given analogous input algorithms for the component theories. Such an algorithm is described in detail in [36]; we give some indications here of how it works.

Below, we consider two theories T_1, T_2 in their respective signatures Σ_1, Σ_2; the two theories are both stably infinite and equality interpolating; moreover, the $SMT(T_1), SMT(T_2)$ problems are decidable and the signatures Σ_1, Σ_2 are disjoint. We also assume the availability of algorithms for T_1 and T_2 that are able not only to compute quantifier-free interpolants, but also the tuples $\underline{v}$ of terms in Definition 4 for the equality-interpolating property. Since the $SMT(T_i)$ problem is decidable for $i = 1, 2$, it is always possible to build an equality-interpolating algorithm by enumeration; in practice, better algorithms can be designed (see [55] for some examples concerning convex theories, see above for non-convex examples regarding some quantifier-eliminating arithmetic theories).

We can restate our problem as follows: We are given a finite set A_0 and a finite set B_0 of $\Sigma_1 \cup \Sigma_2$-ground formulae possibly containing additional free constants. We assume that $A_0 \wedge B_0$ is $T_1 \cup T_2$-unsatisfiable (here, by abuse of notation, we confuse a finite set of formulae with its conjunction). We must find a finite set of ground formulae C (containing at most the free constants occurring both in A_0 and in B_0) such that $A_0 \vdash_{T_1 \cup T_2} C$ and $C \wedge B_0 \vdash_{T_1 \cup T_2} \bot$. Applying standard purification procedures, we can assume that all literals in A_0, B_0 are *pure*, meaning that they cannot contain both an A-strict free constant and a B-strict free constant. Here, we call A-strict (or B-strict) a free constant occurring only in A_0 (or only in B_0); we call it 'shared' if it occurs in both A_0 and B_0. Finally, we call it A-local (or B-local) iff it is either shared or A-strict (or B-strict). A similar terminology is applied to terms, literals, and quantifier-free formulæ; they are said to be A-local, B-local or shared iff they contain only constants that are A-local, B-local or shared, respectively.

The algorithm uses the *metarules* framework introduced in [18]. This framework collects some manipulations that can be freely operated in pairs (A, B) without losing the possibility of computing an interpolant. For instance, if $A \vdash \bigvee_{k=1}^{n} \psi_k$, where the ψ_k are A-local, then it is possible to non-deterministically replace A with $A \cup \{\psi_k\}$, compute all interpolants of $(A \cup \{\psi_k\}, B)$, and then recombine them into an interpolant of (A, B). A long list of metarules is supplied in [18,36]; they are rather simple transformations. Strictly speaking, metarules are not part of an interpolation algorithm; however, if every single transformation of a concrete interpolation algorithm can be reformulated as a combination of metarules, then the algorithm itself is automatically partially correct (that is, it gives a correct answer when it terminates), and only termination requires a proof in order to achieve its total correctness.

The combined interpolation algorithm we are introducing follows this schema. It manipulates A_0, B_0 by applying transformation rules (justified by metarules) that generate a tree labeled by pairs (A, B). In the end, it will be possible to compute, via the input interpolation algorithms, an interpolant out of every leaf; such interpolants will then be recombined to form an interpolant for the original unsatisfiable pair A_0, B_0. While applying the transformation rules, it might happen that some A-strict free constant a 'becomes shared' because an equation $a = t$ explicitly defining it via a shared term is entailed by the current A; the same might happen for a B-strict constant. This 'term-sharing' technique is easily justified by a combination of metarules.

Now, one of the transformation rules simply guesses a Boolean assignment satisfying the current formula A (or B) and adds the corresponding set of literals to A (or B). A Boolean assignment can also guess equalities or disequalities among A-strict (or B-strict) constants, between an A-strict (or B-strict) constant and a shared constant, or between two shared constants. What the assignment cannot do is guess an equality/disequality between an

A-strict and a B-strict constant because no impure literal is tolerated in the interpolant to be built. So, it is assumed by default that A-strict and B-strict constants cannot be equal to each other. When this leads to an inconsistency, this is just because a relation such as

$$A_i \cup B_i \vdash_{T_i} \underline{a} \cap \underline{b} \neq \emptyset$$

holds for $i = 1$ or $i = 2$ (here, $\underline{a} = a_1, \ldots, a_n$ are the A-strict constants, $\underline{b} = b_1, \ldots, b_m$ are the B-strict constants, and A_i, B_i are those among the currently assigned literals that are Σ_i-literals). If this happens, by the equality-interpolating property, there are shared terms $\underline{v} = v_1, \ldots, v_p$ such that

$$A_i \cup B_i \vdash_{T_i} (\underline{a} \cap \underline{v} \neq \emptyset) \vee (\underline{b} \cap \underline{v} \neq \emptyset).$$

Invoking the available interpolation algorithm for T_i, we can compute a ground shared Σ_i-formula θ such that $A \vdash_{T_i} \theta \vee \underline{a} \cap \underline{v} \neq \emptyset$ and $\theta \wedge B \vdash_{T_i} \underline{b} \cap \underline{v} \neq \emptyset$. We choose among $n * p + m * p$ alternatives in order to non-deterministically update A, B in the successor nodes. For the first $n * p$ alternatives, we add some $a_i = v_j$ (for $1 \leq i \leq n$, $1 \leq j \leq p$) to A. For the last $m * p$ alternatives, we add θ to A and some $\{\theta, b_i = v_j\}$ to B (for $1 \leq i \leq m$, $1 \leq j \leq p$). After such updates, the number of the A-strict or of the B-strict free constants decreases because we added an explicitly defining equation in all cases (see the above remark about 'term sharing'). Thus, in the end, it will be possible to assert (explicitly or implicitly) an equality or a disequality for every pair of free constants. Since this is precisely the condition for consistency in the combined Nelson–Oppen procedure [42], this cannot happen because we assumed that $A_0 \wedge B_0$ was $T_1 \cup T_2$-unsatisfiable; hence, every leaf of the tree to be built must contain a contradiction either in the Σ_1-part or in the Σ_2-part of its labeling constraint so that an interpolant can be extracted from every leaf.

4. Non-Disjoint Combinations

Whenever signatures are not disjoint, transfer results are harder to obtain. A crucial notion here is T_0-compatibility [56], which we are going to introduce in the following. This section is based on the results presented in [37,57].

Recall [41] that a *universal* theory T_0 has a *model completion* $T_0^\star$ iff $T_0^\star \supseteq T_0$ is a stronger theory in the same language of T_0 such that: (i) Every model of T_0 can be embedded into a model of $T_0^\star$; and (ii) $T_0^\star$ has quantifier elimination. The definition of a model completion could be suitably extended to theories which are not universal, but we do not need to consider this more general case. Alternative equivalent definitions are possible (for instance, condition (ii) is equivalent to the fact that the union of $T_0^\star$ and of the diagram of a model of T_0 is a complete theory).

Definition 5. *Let T be a theory in the signature Σ and let T_0 be a universal theory in a subsignature $\Sigma_0 \subseteq \Sigma$ such that $T_0 \subseteq T$. We say that T is T_0-*compatible *iff there is a Σ_0-theory $T_0^\star$ such that:*

(i) $T_0 \subseteq T_0^\star$;
(ii) *$T_0^\star$ is a model completion of T_0;*
(iii) *Every model of T can be embedded, as a Σ-structure, into a model of $T \cup T_0^\star$.*

T_0-compatibility is a generalization of stable infiniteness; in fact, a theory T is stably infinite iff it is T_0-compatible, where T_0 is the pure equality theory in the empty signature.

In [56], it is shown that the decidability of the SMT problem transfers from two theories to their non-disjoint combination in case the two theories are both T_0-compatible with respect to a locally finite theory in their shared signature (this result has as a special case the transfer of the decidability of the global consequence relations to fusions of modal logics [58]; see below). More results that replace local finiteness with a so-called 'noetherianity condition' are given in [59].

Strong Amalgamation over a Horn Theory

In [37], two results are given concerning combined quantifier-free interpolation in the case of non-disjoint signatures. We report the second one only, which is easier to formulate and more effective in the applications.

A Σ-theory T is *universal Horn* iff it can be axiomatized via formulæ of the form $A_1 \wedge \cdots \wedge A_n \to B$, where the A_i and B are all atoms(the standard definition of a universal Horn theory would include also the case where B is $\bot$, we disregarded this case for simplicity and because our applications to modal logic do not require it). In purely functional signatures, universal Horn theories axiomatize quasi-varieties.

One important fact is that the categories of models of universal Horn theories are co-complete [60]; hence, in particular, pushouts exist.

Definition 6. *Let T be a theory and let $(\mathcal{M}_1, \mathcal{M}_2, \mathcal{A})$ be a T-fork. A* pushout *of the fork is a triple $(\mathcal{M}, \mu_1, \mu_2)$, where $\mathcal{M}$ is a T-model and $\mu_1 : \mathcal{M}_1 \longrightarrow \mathcal{M}$, $\mu_2 : \mathcal{M}_2 \longrightarrow \mathcal{M}$ are Σ-homomorphisms whose restrictions to the support of $\mathcal{A}$ coincide, such that for every other triple $(\mathcal{M}', \mu'_1, \mu'_2)$ with the same properties, there is a unique homomorphism (called the comparison homomorphism) $\theta : \mathcal{M} \longrightarrow \mathcal{M}'$ such that $\theta \circ \mu_i = \mu'_i$ $(i = 1, 2)$.*

$$\begin{tikzcd}
& & \mathcal{M}' \\
\mathcal{M}_1 \arrow[r, "\mu_1"] \arrow[urr, bend left, "\mu'_1"] & \mathcal{M} \arrow[ur, dotted, "\theta"] & \\
\mathcal{A} \arrow[u, tail] \arrow[r, tail] & \mathcal{M}_2 \arrow[u, "\mu_2"'] \arrow[uur, bend right, "\mu'_2"'] &
\end{tikzcd}$$

If the pushout $(\mathcal{M}, \mu_1, \mu_2)$ of the T-fork $(\mathcal{M}_1, \mathcal{M}_2, \mathcal{A})$ is a T-amalgam (i.e., if μ_1, μ_2 are embeddings), it is called the minimal T-amalgam *of the T-fork.*

Notice that, even when the pushout is a T-amalgam, comparison morphisms need not be injective. This makes the next definition interesting.

Definition 7. *Let T be a theory in a signature Σ; let $T_0 \subseteq T$ be a universal Horn theory in a subsignature $\Sigma_0 \subseteq \Sigma$ with the amalgamation property. We say that T is* T_0-strongly sub-amalgamable *iff every T-fork $(\mathcal{M}_1, \mathcal{M}_2, \mathcal{A})$ has a T-amalgam $\mathcal{M}$ such that the comparison morphism with respect to the minimal T_0-amalgam of the Σ_0-reduct of the T-fork $(\mathcal{M}_1, \mathcal{M}_2, \mathcal{A})$ is injective.*

Notice that strong amalgamation is nothing but T_0-strong amalgamability, where T_0 is the pure equality theory in the empty signature. Thus, the following transfer result is a genuine generalization of Theorem 2:

Theorem 7 ([37])**.** *If T_1, T_2 are both T_0-compatible and T_0-strongly sub-amalgamable (for an amalgamable universal Horn theory T_0 in their common subsignature Σ_0), then so is $T_1 \cup T_2$.*

The above theorem has interesting applications to modal logic. In the following, we let BA be the theory of Boolean algebras. Recall that a Boolean algebra is defined to be a bounded and complemented distributive lattice; since Boolean algebras have a meet-semilattice reduct, it is possible to introduce in them a partial ordering relation $x \leq y$ via the definition $x \sqcap y = x$, where $\sqcap$ is the meet operation.. It is well known [41] that BA has a model completion, which is the theory of atomless Boolean algebras: A Boolean algebra is said to be atomless iff it has no non-zero $\leq$-minimal element.

A *BAO-equational theory* is any theory T whose signature extends the signature of Boolean algebras and whose axioms are all equations and include the Boolean algebra axioms. BAO stands for 'Boolean algebras with operations'. BAO-equational theories arise as algebraic semantics of propositional modal logics [61]; for instance, modal algebras (i.e., Boolean algebras endowed with a unary operator $\Box$ preserving meets and 1) are Lindenbaum algebras of propositional calculi based on the modal system K. However, we do not assume here any 'normality' conditions on the operations of a BAO; hence, BAO are algebraic counterparts of classical modal logics in the sense of [62] (in a classical modal logic, the only assumption made on the modal operators is that the replacement rule for equivalent formulae applies).

The *fusion* of two BAO-equational theories T_1 and T_2 is just their combination $T_1 \cup T_2$ (when speaking of the fusion of T_1 and T_2, we assume that T_1 and T_2 share only the Boolean algebras' operations and no other symbols). This notion of fusion matches with the standard notion of fusion [58] of the modal logics that are counterparts in propositional logic of the algebraic theories T_1, T_2.

Any BAO-equational theory T is BA-compatible; to see this, it is sufficient to show that a model $\mathcal{M}$ of T embeds into a model $\mathcal{M}'$ of T whose Boolean reduct is atomless. This is done by taking the colimit of the chain defined as follows: Let $\mathcal{M}_0$ be $\mathcal{M}$, let $\mathcal{M}_{k+1}$ be $\mathcal{M}_k \times \mathcal{M}_k$, and use the diagonal maps as embeddings $\delta_k : \mathcal{M}_k \longrightarrow \mathcal{M}_{k+1}$.

Thus, in order to apply Theorem 7, we only need to characterize BA-strong subamalgamability. Surprisingly, this is nothing but a well-known notion from the literature.

Definition 8. *We say that a BAO-equational theory T has the* superamalgamation *property [63] iff for every T-fork $(\mathcal{M}_1, \mathcal{M}_2, \mathcal{A})$, there exists a T-amalgam $(\mathcal{M}, \mu_1, \mu_2)$ such that for every $a_1 \in |\mathcal{M}_1|, a_2 \in |\mathcal{M}_2|$ such that $\mu_1(a_1) \leq \mu_2(a_2)$ there exists $a_0 \in |\mathcal{A}|$ such that $a_1 \leq a_0$ holds in $\mathcal{M}_1$ and $a_0 \leq a_2$ holds in $\mathcal{M}_2$.*

Theorem 8 ([37])**.** *A BAO-equational theory T has the superamalgamation property iff it is BA-strongly amalgamable.*

As is well known from [63], the superamalgamation property for varieties of modal algebras, in the case of normal modal logics, corresponds to (the local deducibility version of) the Craig interpolation theorem. Thus, Theorem 8 implies, in particular, a Wolter fusion transfer result [58] of the Craig interpolation theorem for normal modal logics. For non-normal modal logics, superamalgamation corresponds to a strong version of the Craig interpolation theorem (encompassing both the local and the global deducibility versions of it) called the *comprehensive interpolation property* in [37]. Theorem 8 above implies that this comprehensive interpolation property transfers to fusions in the non-normal case, too, as proved in [37].

5. Uniform Interpolants

This section presents results contained in [31,32,38–40,64,65]. First, we analyze a strong form of of quantifier-free interpolation and its relationship with model-completeness [31,32]; we then show that, for convex theories, the same hypotheses allowing the transfer of the existence of ordinary interpolants also allow the transfer of the existence of these stronger interpolants [39,40].

Fix a theory T and an existential formula $\exists \underline{e}\, \phi(\underline{e}, \underline{y})$; a quantifier-free formula $\theta(\underline{y})$ is said to be a T-*cover* [30] (or, simply, a *cover*) of $\exists \underline{e}\, \phi(\underline{e}, \underline{y})$ iff the following two conditions are satisfied: (i) $T \models \exists \underline{e}\, \phi(\underline{e}, \underline{y}) \to \theta(\underline{y})$; (ii) For every formula $\psi(\underline{y}, \underline{z})$ such that $T \models \exists \underline{e}\, \phi(\underline{e}, \underline{y}) \to \psi(\underline{y}, \underline{z})$, we have that $T \models \theta(\underline{y}) \to \psi(\underline{y}, \underline{z})$. Sometimes, the cover $\theta(\underline{y})$ (which is unique up to T-equivalence) is called a *(quantifier-free) uniform interpolant*. The reason for this terminology comes from the fact that an entailment like $T \models \phi(\underline{e}, \underline{y}) \to \psi(\underline{y}, \underline{z})$ is equivalent (by the standard rule for existential quantifier introduction) to $T \models \exists \underline{e}\, \phi(\underline{e}, \underline{y}) \to \psi(\underline{y}, \underline{z})$, hence it is immediately seen that a cover $\theta(\underline{y})$ of $\exists \underline{e}\, \phi(\underline{e}, \underline{y})$ can work as an interpolant for all entailments $T \models \phi(\underline{e}, \underline{y}) \to \psi(\underline{y}, \underline{z})$ (varying all $\psi(\underline{y}, \underline{z})$ for which the entailment holds).

We say that a theory T has *uniform quantifier-free interpolation* iff every existential formula $\exists \underline{e}\, \phi(\underline{e}, \underline{y})$ (equivalently, every primitive formula $\exists \underline{e}\, \phi(\underline{e}, \underline{y})$) has a T-cover.

The following lemma supplies a semantic counterpart to the notion of a cover. What the lemma essentially says is that the cover of $\exists \underline{e}\, \phi(\underline{e}, \underline{y})$ expresses a 'solvability condition' for $\phi(\underline{e}, \underline{y})$ (seen as a kind of system of equations in the parameters $\underline{y}$); if this solvability condition is true, then it is possible to build (maybe in an extended model) a solution for $\phi(\underline{e}, \underline{y})$ and vice versa.

Lemma 1 ([31,32]). *A formula $\psi(\underline{y})$ is a T-cover of $\exists \underline{e}\, \phi(\underline{e}, \underline{y})$ iff it satisfies the following two conditions:*

(i) $T \models \forall \underline{y}\, (\exists \underline{e}\, \phi(\underline{e}, \underline{y}) \to \psi(\underline{y}))$;

(ii) *For every model $\mathcal{M}$ of T and for every tuple of elements $\underline{a}$ from the support of $\mathcal{M}$ such that $\mathcal{M} \models \psi(\underline{a})$, it is possible to find another model $\mathcal{N}$ of T such that $\mathcal{M}$ embeds into $\mathcal{N}$ and $\mathcal{N} \models \exists \underline{e}\, \phi(\underline{e}, \underline{a})$.*

In Section 4, we mentioned the model completion $T^\star$ of a universal theory T; we recall from [41] that $T^\star$ axiomatizes the models of T that are existentially closed for T, i.e., those models $\mathcal{M}$ of T for which an existential formula with parameters in $|\mathcal{M}|$ having a solution in an extension of $\mathcal{M}$, which is also a model of T, has a solution in $\mathcal{M}$ itself. If a theory has uniform interpolation, then every existential formula $\exists \underline{e}\, \phi(\underline{e}, \underline{y})$ has a T-cover, so it is possible to express the solvability condition of $\phi(\underline{e}, \underline{y})$ via the cover. In this way, we can axiomatize existentially closed models; we just say that 'whatever is solvable actually has a solution'. These intuitive considerations show why the following result comes with no surprise.

Theorem 9 ([31,32,66]). *Suppose that T is a universal theory. Then, T has a model completion $T^\star$ iff T has uniform quantifier-free interpolation. If this happens, $T^\star$ is axiomatized by the infinitely many sentences*

$$\forall \underline{y}\, (\psi(\underline{y}) \to \exists \underline{e}\, \phi(\underline{e}, \underline{y})) \tag{19}$$

where $\exists \underline{e}\, \phi(\underline{e}, \underline{y})$ is an existential formula and ψ is a T-cover of it.

5.1. Uniform Interpolants in $\mathcal{EUF}$

Whereas it is clear that theories enjoying quantifier elimination also have uniform interpolation, it is less evident whether other theories used in verification have covers or not. $\mathcal{EUF}$ is the typical theory used in verification that is not a theory axiomatizing arithmetic data and that does not enjoy quantifier elimination. That is why investigating uniform interpolation in $\mathcal{EUF}$ can be interesting. In fact, $\mathcal{EUF}$ does enjoy uniform interpolation; the result was stated on various occasions in the literature (including, e.g., [30]), but the first proofs were only published in the conference paper [31] and its journal version [32]. Alternative proofs are reported in [38,65]. Actually, these papers contain three different algorithms for computing uniform interpolants in $\mathcal{EUF}$. In this subsection, we only report the first algorithm from [38,64,65], which is the simplest one to explain.

The algorithm is based on transformation rules. We first need some definitions (for simplicity, we assume that the signature is functional).

A *flat literal* is a literal included in the following list:

$$f(a_1, \dots, a_n) = b, \quad a_1 = a_2, \quad a_1 \neq a_2 \tag{20}$$

where $a_1, \dots, a_n$ and b are (not necessarily distinct) variables or constants. A formula is flat iff all literals occurring in it are flat; flat terms are terms occurring in the literals listed above in (20).

An *explicit definition via a directed acyclic graph* (abbreviated as a DAG-definition, or simply as a DAG) is any formula $\mathtt{ExplDef}(\underline{y}, \underline{z})$ of the following form (where $\underline{y} := y_1 \dots, y_n$):

$$\bigwedge_{i=1}^{n} (y_i = f_i(y_1, \dots, y_{i-1}, \underline{z}))\,.$$

Thus, $\texttt{ExplDef}(\underline{y}, \underline{z})$ provides an *explicit definition* of the $\underline{y}$ in terms of the parameters $\underline{z}$. Given such a DAG-definition $\texttt{ExplDef}(\underline{y}, \underline{z})$, we can, in fact, associate to it a substitution σ so that a formula such as

$$\exists \underline{y}\ (\texttt{ExplDef}(\underline{y}, \underline{z}) \land \Phi(\underline{y}, \underline{z})) \tag{21}$$

is equivalent to $\Phi(\sigma(\underline{y}), \underline{z})$. The formula $\Phi(\sigma(\underline{y}), \underline{z})$ is said to be the *unravelling* of (21); notice that computing such an unravelling by explicitly performing the required substitutions causes an exponential blow-up.

We want to compute the cover of a primitive formula $\exists \underline{e}\, \phi(\underline{e}, \underline{z})$; we can assume without loss of generality that *the constraint* $\phi(\underline{e}, \underline{z})$ *is flat*. To see this, it is sufficient to apply (as a pre-processing step) the well-known Congruence Closure Transformations, as explained, e.g., in [67] (these transformations have a linear cost).

The algorithm manipulates formulae in the following format:

$$\exists \underline{y}\ (\texttt{ExplDef}(\underline{y}, \underline{z}) \land \Phi(\underline{y}, \underline{z}) \land \exists \underline{e}\ \Psi(\underline{e}, \underline{y}, \underline{z})) \tag{22}$$

where $\texttt{ExplDef}(\underline{y}, \underline{z})$ is a DAG and Φ, Ψ are flat constraints (notice that the $\underline{e}$ do not occur in Φ). When writing formulae such as (22), we usually omit the existential quantifiers $\exists \underline{y}$ and $\exists \underline{e}$ for brevity.

Initially, $\texttt{ExplDef}$ and Φ are the empty conjunction. In (22), the variables $\underline{z}$ are called *parameter* variables, the variables $\underline{y}$ are called *(explicitly) defined* variables, and the variables $\underline{e}$ are called *(truly) quantified* variables. The algorithm does not modify the variables $\underline{z}$; on the other hand, it might cause some quantified variable to disappear or to be renamed as a defined variable. Below, the letters $a, b, \ldots$ range over $\underline{e} \cup \underline{y} \cup \underline{z}$.

Definition 9. *A term t (or a literal L) is $\underline{e}$-free when there is no occurrence of any of the variables $\underline{e}$ in t (or in L). Two flat terms t, u of the kinds*

$$t := f(a_1, \ldots, a_n) \qquad u := f(b_1, \ldots, b_n) \tag{23}$$

are said to be compatible *iff, for every $i = 1, \ldots, n$, either a_i is identical to b_i or both a_i and b_i are $\underline{e}$-free. The* difference set *of two compatible terms as above is the set of disequalities $a_i \neq b_i$, where a_i is not identical to b_i.*

The algorithm (taken from [38,64,65]) applies the rules below in any order, except the last one, which has lower priority. The last rule splits the execution of the algorithm into several branches; each branch will produce a different disjunct in the output formula.

(1) *Simplification Rules*:

- (1.0) If an atom such as $t = t$ belongs to Ψ, just remove it; if a literal such as $t \neq t$ occurs somewhere, delete Ψ, replace Φ with $\bot$, and stop;
- (1.i) If t is not a variable and Ψ contains both $t = a$ and $t = b$, remove the former and replace it with $a = b$.
- (1.ii) If Ψ contains $e_i = e_j$ with $i > j$, remove it and replace e_i with e_j everywhere.

(2) *DAG Update Rule*: If Ψ contains $e_i = t(\underline{y}, \underline{z})$, remove it, rename e_i as y_j everywhere (for fresh y_j), and add $y_j = t(\underline{y}, \underline{z})$ to $\texttt{ExplDef}(\underline{y}, \underline{z})$. More formally:

$$\texttt{ExplDef}(\underline{y}, \underline{z}) \land \Phi(\underline{y}, \underline{z}) \land \Big(\Psi(\underline{e}, e_i, \underline{y}, \underline{z}) \land e_i = t(\underline{y}, \underline{z})\Big)$$

$$\Downarrow$$

$$\Big(\texttt{ExplDef}(\underline{y}, \underline{z}) \land y_j = t(\underline{y}, \underline{z})\Big) \land \Phi(\underline{y}, \underline{z}) \land \Psi(\underline{e}, y_j, \underline{y}, \underline{z})$$

(3) $\boxed{\underline{e}\text{-}\textit{Free Literal Rule}\text{:}}$ If Ψ contains a literal $L(\underline{y}, \underline{z})$, move it to $\Phi(\underline{y}, \underline{z})$. More formally:

$$\texttt{ExplDef}(\underline{y}, \underline{z}) \wedge \Phi(\underline{y}, \underline{z}) \wedge \Big(\Psi(\underline{e}, \underline{y}, \underline{z}) \wedge L(\underline{y}, \underline{z})\Big)$$

$$\Downarrow$$

$$\texttt{ExplDef}(\underline{y}, \underline{z}) \wedge \Big(\Phi(\underline{y}, \underline{z}) \wedge L(\underline{y}, \underline{z})\Big) \wedge \Psi(\underline{e}, \underline{y}, \underline{z})$$

(4) $\boxed{\textit{Splitting Rule}\text{:}}$ If Ψ contains a pair of atoms $t = a$ and $u = b$, where t and u are compatible flat terms as in (23) (thus, in particular, t and u are of the kinds $f(a_1, \dots, a_n)$ and $f(b_1, \dots, b_n)$, respectively) , and no disequality from the difference set of t, u belongs to Φ, then apply one of the following alternatives:

(4.0) Remove from Ψ the atom $f(b_1, \dots, b_n) = b$, add to Ψ the atom $a = b$, and add to Φ all equalities $a_i = b_i$ such that $a_i \neq b_i$ is in the difference set of t, u;

(4.1) Add to Φ one of the disequalities from the difference set of t, u (notice that the difference set cannot be empty; otherwise, Rule (1.i) applies).

Theorem 10 ([38,64,65])**.** *Suppose that we apply the above algorithm to the primitive formula $\exists \underline{e}(\phi(\underline{e}, \underline{z}))$ and that the algorithm terminates with its branches in the states*

$$\texttt{ExplDef}_1(\underline{y}, \underline{z}) \wedge \Phi_1(\underline{y}_1, \underline{z}) \wedge \Psi_1(\underline{e}_1, \underline{y}_1, \underline{z}), \quad \dots, \quad \texttt{ExplDef}_k(\underline{y}, \underline{z}) \wedge \Phi_k(\underline{y}_k, \underline{z}) \wedge \Psi_k(\underline{e}_k, \underline{y}_k, \underline{z})$$

Then, the cover of $\exists \underline{e}(\phi(\underline{e}, \underline{z}))$ in $\mathcal{EUF}$ is the disjunction of the unravellings of the formulæ

$$\exists \underline{y}_i\, (\texttt{ExplDef}_i(\underline{y}, \underline{z}) \wedge \Phi_i(\underline{y}_i, \underline{z})) \tag{24}$$

while varying $i = 1, \dots, k$.

The proof (shown in [38,65]) is based on Lemma 1 and essentially shows that if we conjoin the Robinson Diagram of a model satisfying $\texttt{ExplDef}_i(\underline{y}, \underline{z}) \wedge \Phi_i(\underline{y}_i, \underline{z})$ (relatively to a certain assignment to the variables $\underline{y}_i, \underline{z}$) with $\Psi_i(\underline{e}_i, \underline{y}_i, \underline{z})$, we get a canonical rewrite system (under a suitable reduction ordering; see [68] for information on rewrite systems).

Example 6. *This example is analyzed in [38,64,65]. Let us compute a cover of the formula $\exists e_0\, (g(z_4, e_0) = z_0 \wedge f(z_2, e_0) = g(z_3, e_0) \wedge h(f(z_1, e_0)) = z_0)$. We first need to flatten it; this produces the set of literals*

$$g(z_4, e_0) = z_0 \wedge e_1 = f(z_2, e_0) \wedge e_1 = g(z_3, e_0) \wedge e_2 = f(z_1, e_0) \wedge h(e_2) = z_0 \tag{25}$$

where we have two newly introduced variables e_1, e_2 to be eliminated, too. After applying Splitting (4.0), the equality $g(z_3, e_0) = e_1$ is removed and the new equalities $z_3 = z_4$, $e_1 = z_0$ are introduced. Now, (2) renames e_1 as y_1 by (2). We apply (4.0) again; this removes $f(z_1, e_0) = e_2$ and adds the equalities $z_1 = z_2$, $e_2 = y_1$; moreover, the variable e_2 is renamed as y_2. We can now apply (3) to the literal $h(y_2) = z_0$. The branch terminates with $y_1 = z_0 \wedge y_2 = y_1 \wedge z_1 = z_2 \wedge z_3 = z_4 \wedge h(y_2) = z_0 \wedge f(z_2, e_0) = y_1 \wedge g(z_4, e_0) = z_0$. This gives $z_1 = z_2 \wedge z_3 = z_4 \wedge h(z_0) = z_0$ as a first disjunct of the uniform interpolant. The other branches give $z_1 = z_2 \wedge z_3 \neq z_4$, $z_1 \neq z_2 \wedge z_3 = z_4$, and $z_1 \neq z_2 \wedge z_3 \neq z_4$ as further disjuncts, so that our cover turns out to be logically equivalent to $z_1 = z_2 \wedge z_3 = z_4 \to h(z_0) = z_0$.

The above algorithm has exponential complexity (the branches have quadratic size); notice, however, that, if the signature only contains unary function symbols, there is no need to apply the Splitting Rule, and hence, the complexity is polynomial; the case of a signature with only unary function symbols is important in the applications to data-aware verification because it allows the formalization of read-only databases with primary and foreign keys [35].

5.2. Combined Uniform Interpolants

We now investigate combined uniform interpolants by starting from the convex case and by showing the algorithm presented in [39,40]. Let us fix a *convex, stably infinite, equality-interpolating universal theory admitting a model completion*. Let T be such a theory, let Σ be its signature, and let $T^\star$ be its model completion. Consider a conjunction of Σ-literals $\phi(\underline{x}, \underline{y})$, where $\underline{y} = y_1, \ldots, y_n$ (recall that the tuple $\underline{x}$ is disjoint from the tuple $\underline{y}$ according to the conventions from Section 2).

For $i = 1, \ldots, n$, we let the formula $\texttt{ImplDef}^T_{\phi, y_i}(\underline{x})$ be the quantifier-free formula equivalent in $T^\star$ to the formula

$$\forall \underline{y}\, \forall \underline{y}' (\phi(\underline{x}, \underline{y}) \wedge \phi(\underline{x}, \underline{y}') \to y_i = y'_i) \tag{26}$$

where the $\underline{y}'$ are renamed copies of the $\underline{y}$. The following lemma (taken from [39,40]) comes from the primitive Beth definability property (recall the paragraph following Proposition 1):.

Lemma 2. *Let $L_{i1}(\underline{x}) \vee \cdots \vee L_{ik_i}(\underline{x})$ be the disjunctive normal form (DNF) of $\texttt{ImplDef}^T_{\phi, y_i}(\underline{x})$. Then, for every $j = 1, \ldots, k_i$, there is a $\Sigma(\underline{x})$-term $t_{ij}(\underline{x})$ such that*

$$T \vdash L_{ij}(\underline{x}) \wedge \phi(\underline{x}, \underline{y}) \to y_i = t_{ij} \ . \tag{27}$$

The above lemma is the key technical ingredient for the proof of the following result.

Theorem 11 ([39,40])**.** *Let T_1, T_2 be convex, stably infinite, equality-interpolating, universal theories over disjoint signatures with uniform quantifier-free interpolation. Then, $T_1 \cup T_2$ has uniform quantifier-free interpolation.*

We now present the algorithm from [39,40] to compute covers in $T_1 \cup T_2$ when the hypotheses of the above theorem are satisfied and the $SMT(T_1), SMT(T_2)$ satisfiability problems are decidable. We show how compute the cover of a primitive formula $\exists \underline{e}\, \phi(\underline{e}, \underline{z})$, where we freely assume that the literals in ϕ are all flat: if we let Σ_1 to be the signature of T_1 and Σ_2 to be the signature of T_2, flatness means in particular that such literals are either Σ_1-literals or Σ_2-literals or both (the latter can obviously be the case only for equalities or negated equalities involving variables). The idea behind the algorithm is that the input cover algorithms can be separately applied, once all potential definability phenomena have been identified.

A *working formula* is a formula of the kind

$$\exists \underline{z}\, (\texttt{ExplDef}(\underline{z}, \underline{x}) \wedge \exists \underline{e}\, (\psi_1(\underline{x}, \underline{z}, \underline{e}) \wedge \psi_2(\underline{x}, \underline{z}, \underline{e}))) \ , \tag{28}$$

where $\texttt{ExplDef}(\underline{z}, \underline{x})$ is a DAG, ψ_1 is a conjunction of Σ_1-literals, and ψ_2 is a conjunction of Σ_2-literals. We assume that ψ_1, ψ_2 in a working formula (28) always contain the literals $e_i \neq e_j$ (for distinct e_i, e_j from $\underline{e}$) as a conjunct; this can be forced at the initialization stage by making a case-split followed by replacements of equals by equals. Contrary to what we did in the $\mathcal{EUF}$ case above, here, we do not need to separate the literals that do not contain the truly existential variables $\underline{e}$ from the other ones.

A working formula such as (28) is said to be *terminal* iff, for every existential variable $e_i \in \underline{e}$, we have that

$$T_1 \vdash \psi_1 \to \neg\texttt{ImplDef}^{T_1}_{\psi_1, e_i}(\underline{x}, \underline{z}) \ \text{ and } \ T_2 \vdash \psi_2 \to \neg\texttt{ImplDef}^{T_2}_{\psi_2, e_i}(\underline{x}, \underline{z}) \ . \tag{29}$$

Intuitively, in a terminal working formula, all variables that are not parameters are either explicitly definable or recognized not to be implicitly definable by both theories. Notice that the validity tests for the implications (29) can be effectively discharged using the quantifier-free satisfiability procedures in T_1, T_2.

We first observe (see [39,40] for details) that every working formula is equivalent (modulo $T_1 \cup T_2$) to a disjunction of terminal working formulæ. Such a disjunction of terminal working formulæ can be computed as follows: One exhaustively applies the following transformations in all possible ways (the output is the disjunction of the different outcomes).

(1) Update ψ_1 by adding to it a disjunct from the DNF of $\bigwedge_{e_i \in \underline{e}} \neg \mathtt{ImplDef}^{T_1}_{\psi_1, e_i}(\underline{x}, \underline{z})$ and ψ_2 by adding to it a disjunct from the DNF of $\bigwedge_{e_i \in \underline{e}} \neg \mathtt{ImplDef}^{T_2}_{\psi_2, e_i}(\underline{x}, \underline{z})$;

(2.i) Select $e_i \in \underline{e}$ and $h \in \{1, 2\}$; then, update ψ_h by adding to it a disjunct L_{ij} from the DNF of $\mathtt{ImplDef}^{T_h}_{\psi_h, e_i}(\underline{x}, \underline{z})$; the equality $e_i = t_{ij}$ (where t_{ij} is the term mentioned in Lemma 2) is added to $\mathtt{ExplDef}(\underline{z}, \underline{x})$; the variable e_i becomes, in this way, part of the defined variables.

To conclude, we need the final fact (again, shown in [39,40]) that the cover of a working Formula (28) that is terminal is given by the unravelling of the explicit definitions of the variables $\underline{z}$ from the formula

$$\exists \underline{z}\ (\mathtt{ExplDef}(\underline{z}, \underline{x}) \wedge \theta_1(\underline{x}, \underline{z}) \wedge \theta_2(\underline{x}, \underline{z})) \tag{30}$$

where $\theta_1(\underline{x}, \underline{z})$ is the T_1-cover of $\exists \underline{e}\, \psi_1(\underline{x}, \underline{z}, \underline{e})$ and $\theta_2(\underline{x}, \underline{z})$ is the T_2-cover of $\exists \underline{e}\, \psi_2(\underline{x}, \underline{z}, \underline{e})$.

A remarkable corollary of the above theorem says that existence of uniform interpolants is preserved when adding free function symbols to a convex, stably infinite, equality-interpolating, universal theory with uniform interpolants (this is because the combination with $\mathcal{EUF}$ enjoys the hypotheses of Theorem 11). Unfortunately, the convexity hypothesis is indispensable for this result to hold, as the following counterexample from [39,40] shows.

We take as T_1 the integer difference logic $\mathcal{IDL}$ of Example 2; notice that this theory is stably infinite, universal, and has quantifier elimination (thus, it coincides with its own model completion). This theory is not convex; however, it is equality interpolating, as seen in Section 3 above. As T_2, we take $\mathcal{EUF}(\Sigma_f)$, where Σ_f has just one unary free function symbol f (this f is supposed not to belong to the signature of T_1).

Proposition 2 ([39,40])**.** *Let T_1, T_2 be as above; the formula*

$$\exists e\ (0 < e \wedge e < x \wedge f(e) = 0) \tag{31}$$

does not have a cover in $T_1 \cup T_2$.

The counterexample still applies when replacing integer difference logic with linear integer arithmetics.

6. Conclusions

We investigated transfer results concerning the existence of quantifier-free (ordinary and uniform) interpolants to combined first-order theories. The investigation used semantic and model-theoretic tools in an essential way in order to obtain appropriate conceptualizations for justifying concrete algorithms.

Some problems are left open; in particular, the results concerning the case of non-disjoint signatures are far from being exhaustive. Indeed, in the case of non-disjoint signatures, sufficient conditions for the transfer of uniform interpolants are completely missing.

Advanced combination problems tend to be rather difficult in nature; however, very often, applications show unexpectedly interesting research perspectives that are worth pursuing. For instance, in [40], a strong result (working only under the stable infiniteness hypothesis) for the transfer of existence of uniform interpolants is obtained in the case of special many-sorted disjoint signature combinations (called 'tame combinations') arising in the area of verification of data-aware processes [34,35,69].

Applications are also important for testing the feasibility of the algorithms suggested by theoretical research in concrete implementations. As briefly mentioned in the introduction, frameworks for the verification of data-aware processes [35,69,70] provide a particularly interesting setting where (combined) uniform interpolation plays a crucial role from the theoretical, the methodological/algorithmic, and the operational perspectives (see [71] for an exhaustive introduction to this topic). In this context, complex dynamic systems that can interact with a persistent data storage are verified against some property of interest via sophisticated techniques based on SMT-solving and on automated reasoning; specifically, the presence of the 'data' component, which is usually represented as relational databases that the 'process' component can query and update, requires the development of suitable techniques for eliminating (to some extent) quantifiers binding variables that range over the content of such databases. This task can be effectively and efficiently performed by computing (combined) uniform interpolants [31,32,39,40]. This motivated the implementation of algorithms for computing combined uniform interpolants in the state-of-the-art MCMT model checker [72]. We demonstrated the feasibility of this approach in [71] by testing MCMT against a benchmark of concrete data-aware processes, and we showed in [69–71,73,74] how these techniques turn out to be extremely useful for developing operational verification frameworks for modeling and verifying business processes enriched with concrete data that emerge in real-world scenarios and in business process management [75] within contemporary organizations.

Author Contributions: All authors contributed equally to this work. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Craig, W. Three uses of the Herbrand-Gentzen theorem in relating model theory and proof theory. *J. Symb. Log.* **1957**, *22*, 269–285. [CrossRef]
2. McMillan, K.L. Interpolation and SAT-Based Model Checking. In *International Conference on Computer Aided Verification*; Springer: Berlin/Heidelberg, Germany, 2003; Volume 2725, pp. 1–13. [CrossRef]
3. McMillan, K.L. Applications of Craig Interpolation to Model Checking. In Proceedings of the CSL, Karpacz, Poland, 20–24 September 2004; Springer: Berlin/Heidelberg, Germany, 2004; Volume 3210, pp. 22–23. [CrossRef]
4. McMillan, K.L. Lazy Abstraction with Interpolants. In Proceedings of the CAV, Seattle, WA, USA, 17–20 August 2006; Springer: Berlin/Heidelberg, Germany, 2006; Volume 4144, pp. 123–136. [CrossRef]
5. Jhala, R.; McMillan, K.L. Interpolant-Based Transition Relation Approximation. In Proceedings of the CAV, Scotland, UK, 6–10 July 2005; Springer: Berlin/Heidelberg, Germany, 2005; Volume 3576, pp. 39–51.
6. McMillan, K.L. Quantified Invariant Generation Using an Interpolating Saturation Prover. In Proceedings of the TACAS, Budapest, Hungary, 29 March–6 April 2008; Springer: Berlin/Heidelberg, Germany, 2008; Volume 4963, pp. 413–427.
7. Alberti, F.; Bruttomesso, R.; Ghilardi, S.; Ranise, S.; Sharygina, N. SAFARI: SMT-Based Abstraction for Arrays with Interpolants. In Proceedings of the CAV, Berkeley, CA, USA, 7–13 July 2012; Springer: Berlin/Heidelberg, Germany, 2012; Volume 7358, pp. 679–685. [CrossRef]
8. Alberti, F.; Ghilardi, S.; Sharygina, N. Booster: An Acceleration-Based Verification Framework for Array Programs. In *International Symposium on Automated Technology for Verification and Analysis*; Springer: Berlin/Heidelberg, Germany, 2014; Volume 8837, pp. 18–23. [CrossRef]
9. Vizel, Y.; Gurfinkel, A. Interpolating Property Directed Reachability. In Proceedings of the CAV, Vienna, Austria, 18–22 July 2014; Springer: Berlin/Heidelberg, Germany, 2014; Volume 8559, pp. 260–276. [CrossRef]
10. Krishnan, H.G.V.; Vizel, Y.; Ganesh, V.; Gurfinkel, A. Interpolating Strong Induction. In Proceedings of the CAV, New York, NY, USA, 15–18 July 2019; Springer: Berlin/Heidelberg, Germany, 2019; Volume 11562, pp. 367–385. [CrossRef]
11. McMillan, K.L. An interpolating theorem prover. *Theor. Comput. Sci.* **2005**, *345*, 101–121. [CrossRef]
12. Rybalchenko, A.; Sofronie-Stokkermans, V. Constraint Solving for Interpolation. In Proceedings of the VMCAI, Nice, France, 14–16 January 2007; Springer: Berlin/Heidelberg, Germany, 2007; Volume 4349, pp. 346–362.
13. Sofronie-Stokkermans, V. Interpolation in Local Theory Extensions. *Log. Methods Comput. Sci.* **2008**, *4*. [CrossRef]

14. Jain, H.; Clarke, E.M.; Grumberg, O. Efficient Craig Interpolation for Linear Diophantine (Dis)Equations and Linear Modular Equations. In Proceedings of the CAV, Princeton, NJ, USA, 7–14 July 2008; Springer: Berlin/Heidelberg, Germany, 2008; Volume 5123, pp. 254–267.
15. Cimatti, A.; Griggio, A.; Sebastiani, R. Efficient Interpolant Generation in Satisfiability Modulo Theories. In Proceedings of the TACAS, Budapest, Hungary, 29 March–6 April 2008; Springer: Berlin/Heidelberg, Germany, 2008; Volume 4963, pp. 397–412.
16. Fuchs, A.; Goel, A.; Grundy, J.; Krstic, S.; Tinelli, C. Ground Interpolation for the Theory of Equality. In Proceedings of the TACAS, York, UK, 22–29 March 2009; Springer: Berlin/Heidelberg, Germany, 2009; Volume 5505, pp. 413–427.
17. Cimatti, A.; Griggio, A.; Sebastiani, R. Interpolant Generation for UTVPI. In Proceedings of the CADE-22, Montreal, QC, Canada, 2–7 August 2009; Springer: Berlin/Heidelberg, Germany, 2009; Volume 5663, pp. 167–182.
18. Bruttomesso, R.; Ghilardi, S.; Ranise, S. Quantifier-Free Interpolation of a Theory of Arrays. *Log. Methods Comput. Sci.* **2012**, *8*. [CrossRef]
19. Totla, N.; Wies, T. Complete Instantiation-Based Interpolation. *J. Autom. Reason.* **2016**, *57*, 37–65. [CrossRef]
20. Ghilardi, S.; Gianola, A.; Kapur, D. Interpolation and Amalgamation for Arrays with MaxDiff. In Proceedings of the FOSSACS, Luxembourg, 27 March–1 April 2021; Springer: Berlin/Heidelberg, Germany, 2021; Volume 12650, pp. 268–288. [CrossRef]
21. Shavrukov, V. *Subalgebras of Diagonalizable Algebras of Theories Containing Arithmetic*; Dissertationes Mathematicae; Polska Akademia Nauk: Warsaw, Poland, 1993; Volume CCCXXIII.
22. Visser, A. Uniform interpolation and layered bisimulation. In *Gödel 96: Logical Foundations on Mathematics, Computer Science and Physics—Kurt Gödel's Legacy*; Hájek, P., Ed.; Springer: Berlin/Heidelberg, Germany, 1996.
23. Ghilardi, S.; Zawadowski, M. *Sheaves, Games, and MODEL Completions*; Trends in Logic—Studia Logica Library; Kluwer Academic Publishers: Dordrecht, The Netherlands, 2002; Volume 14. [CrossRef]
24. Ghilardi, S. An Algebraic Theory of Normal Forms. *Ann. Pure Appl. Log.* **1995**, *71*, 189–245. [CrossRef]
25. Bílková, M. Uniform Interpolation and Propositional Quantifiers in Modal Logics. *Stud. Log.* **2007**, *85*, 1–31. [CrossRef]
26. van Gool, S.J.; Metcalfe, G.; Tsinakis, C. Uniform interpolation and compact congruences. *Ann. Pure Appl. Logic* **2017**, *168*, 1927–1948. [CrossRef]
27. Kowalski, T.; Metcalfe, G. Uniform interpolation and coherence. *Ann. Pure Appl. Logic* **2019**, *170*, 825–841 . [CrossRef]
28. Metcalfe, G.; Reggio, L. Model Completions for Universal Classes of algebras: Necessary and sufficient conditions. *arXiv* **2021**, arXiv:2102.01426v1.
29. Kapur, D. Nonlinear Polynomials, Interpolants and Invariant Generation for System Analysis. In Proceedings of the SC-Square 2017 (Co-Located with ISSAC 2017), CEUR Workshop Proceedings, Kaiserslautern, Germany, 29 July 2017; Volume 1974.
30. Gulwani, S.; Musuvathi, M. Cover Algorithms and Their Combination. In Proceedings of the ESOP 2008 (Held as Part of ETAPS 2008), Budapest, Hungary, 29 March–6 April 2008; Springer: Berlin/Heidelberg, Germany, 2008; Volume 4960, pp. 193–207. [CrossRef]
31. Calvanese, D.; Ghilardi, S.; Gianola, A.; Montali, M.; Rivkin, A. Model Completeness, Covers and Superposition. In Proceedings of the CADE 2019, Natal, Brazil, 27–30 August 2019; Springer: Berlin/Heidelberg, Germany, 2019; Volume 11716, pp. 142–160. [CrossRef]
32. Calvanese, D.; Ghilardi, S.; Gianola, A.; Montali, M.; Rivkin, A. Model Completeness, Uniform Interpolants and Superposition Calculus. *J. Autom. Reason.* **2021**, *65*, 941–969. [CrossRef]
33. Calvanese, D.; Ghilardi, S.; Gianola, A.; Montali, M.; Rivkin, A. From Model Completeness to Verification of Data Aware Processes. In *Description Logic, Theory Combination, and All That*; Springer: Cham, Switzerland, 2019; Volume 11560, pp. 212–239. [CrossRef]
34. Calvanese, D.; Ghilardi, S.; Gianola, A.; Montali, M.; Rivkin, A. Verification of Data-Aware Processes: Challenges and Opportunities for Automated Reasoning. In Proceedings of the ARCADE 2019 EPTCS, Natal, Brazil, 26 August 2019; Volume 311, pp. 53–58. [CrossRef]
35. Calvanese, D.; Ghilardi, S.; Gianola, A.; Montali, M.; Rivkin, A. SMT-based verification of data-aware processes: A model-theoretic approach. *Math. Struct. Comput. Sci.* **2020**, *30*, 271–313. [CrossRef]
36. Bruttomesso, R.; Ghilardi, S.; Ranise, S. Quantifier-free interpolation in combinations of equality interpolating theories. *ACM Trans. Comput. Log.* **2014**, *15*, 1–34. [CrossRef]
37. Ghilardi, S.; Gianola, A. Modularity results for interpolation, amalgamation and superamalgamation. *Ann. Pure Appl. Log.* **2018**, *169*, 731–754. [CrossRef]
38. Ghilardi, S.; Gianola, A.; Kapur, D. Compactly Representing Uniform Interpolants for EUF using (conditional) DAGS. *arXiv* **2020**, arXiv:2002.09784.
39. Calvanese, D.; Ghilardi, S.; Gianola, A.; Montali, M.; Rivkin, A. Combined Covers and Beth Definability. In Proceedings of the IJCAR2020, Paris, France, 1–4 July 2020; Springer: Berlin/Heidelberg, Germany, 2020; Volume 12166, pp. 181–200. [CrossRef]
40. Calvanese, D.; Ghilardi, S.; Gianola, A.; Montali, M.; Rivkin, A. Combination of Uniform Interpolants via Beth Definability. *J. Autom. Reason.* **2022**, *under review*.
41. Chang, C.C.; Keisler, H.J. *Model Theory*, 3rd ed.; North-Holland Publishing Co.: Amsterdam, The Netherlands; London, UK, 1990.
42. Nelson, G.; Oppen, D.C. Simplification by Cooperating Decision Procedures. *ACM Trans. Program. Lang. Syst.* **1979**, *1*, 245–257. [CrossRef]
43. Tinelli, C.; Harandi, M.T. A New Correctness Proof of the {Nelson-Oppen} Combination Procedure. In Proceedings of the FroCoS 1996, Munich, Germany, 26–29 March 1996; Kluwer Academic Publishers: Dordrecht, The Netherlands, 1996; Volume 3, pp. 103–119.

44. Bonacina, M.P.; Ghilardi, S.; Nicolini, E.; Ranise, S.; Zucchelli, D. Decidability and Undecidability Results for Nelson-Oppen and Rewrite-Based Decision Procedures. In Proceedings of the IJCAR 2006, Seattle, WA, USA, 17–20 August 2006; Springer: Berlin/Heidelberg, Germany, 2006; Volume 4130, pp. 513–527. [CrossRef]
45. Bonacina, M.P.; Fontaine, P.; Ringeissen, C.; Tinelli, C. Theory Combination: Beyond Equality Sharing. In *Description Logic, Theory Combination, and All That*; Springer: Berlin/Heidelberg, Germany, 2019; Volume 11560, pp. 57–89. [CrossRef]
46. Sheng, Y.; Zohar, Y.; Ringeissen, C.; Reynolds, A.; Barrett, C.W.; Tinelli, C. Politeness and Stable Infiniteness: Stronger Together. In Proceedings of the CADE 2021, Pittsburgh, PA, USA, 12–15 July 2021; Springer: Berlin/Heidelberg, Germany, 2021; Volume 12699, pp. 148–165. [CrossRef]
47. Kiss, E.W.; Márki, L.; Pröhle, P.; Tholen, W. Categorical algebraic properties. A compendium on amalgamation, congruence extension, epimorphisms, residual smallness, and injectivity. *Studia Sci. Math. Hungar.* **1982**, *18*, 79–140.
48. Bacsich, P.D. Amalgamation properties and interpolation theorems for equational theories. *Algebra Universalis* **1975**, *5*, 45–55. [CrossRef]
49. McCarthy, J. Towards a Mathematical Science of Computation. In *IFIP Congress*; Springer: Dordrecht, The Netherlands, 1962; pp. 21–28.
50. Kapur, D.; Majumdar, R.; Zarba, C.G. Interpolation for data structures. In Proceedings of the SIGSOFT-FSE 2006, Portland, OR, USA, 5–11 November 2006; pp. 105–116. [CrossRef]
51. Oppen, D.C. Reasoning about Recursively Defined Data Structures. *J. ACM* **1980**, *27*, 403–411. [CrossRef]
52. Mal'cev, A.I. Axiomatizable Classes of Locally Free Algebras of Certain Types. *Sibirsk. Mat. Ž.* **1962**, *3*, 729–743.
53. Enderton, H.B. *A Mathematical Introduction to Logic*; Academic Press: New York, NY, USA; London, UK, 1972.
54. Brillout, A.; Kroening, D.; Rümmer, P.; Wahl, T. Beyond Quantifier-Free Interpolation in Extensions of Presburger Arithmetic. In Proceedings of the VMCAI 2011, Austin, TX, USA, 23–25 January 2011; Springer: Berlin/Heidelberg, Germany, 2011; Volume 6538, pp. 88–102. [CrossRef]
55. Yorsh, G.; Musuvathi, M. A Combination Method for Generating Interpolants. In Proceedings of the CADE 2005, Tallinn, Estonia, 22–27 July 2005; Springer: Berlin/Heidelberg, Germany, 2005; Volume 3632, pp. 353–368. [CrossRef]
56. Ghilardi, S. Model-Theoretic Methods in Combined Constraint Satisfiability. *J. Autom. Reason.* **2004**, *33*, 221–249. [CrossRef]
57. Ghilardi, S.; Gianola, A. Interpolation, Amalgamation and Combination (The Non-disjoint Signatures Case). In Proceedings of the FroCoS 2017, Brasilia, Brazil, 27–29 September 2017; Springer: Berlin/Heidelberg, Germany, 2017; Volume 10483, pp. 316–332. [CrossRef]
58. Wolter, F. Fusions of modal logics revisited. In *Advances in Modal Logic*; CSLI Lecture Notes; World Scientific: London, UK, 1998; pp. 361–379.
59. Ghilardi, S.; Nicolini, E.; Zucchelli, D. A comprehensive combination framework. *ACM Trans. Comput. Log.* **2008**, *9*, 1–54. [CrossRef]
60. Adamek, J.; Rosicky, J. *Locally Presentable and Accessible Categories*; Cambridge University Press: Cambridge, UK, 1994.
61. Rasiowa, H. *An Algebraic Approach to Non Classical Logics*; North-Holland: Amsterdam, The Netherlands, 1974.
62. Segerberg, K. *An Essay in Classical Modal Logic*; Filosofiska Studier; Uppsala Universitet: Uppsala, Sweden, 1971; Volume 13.
63. Maksimova, L.L. Interpolation theorems in modal logics and amalgamable varieties of topological Boolean algebras. *Algebra Log.* **1979**, *18*, 556–586. [CrossRef]
64. Ghilardi, S.; Gianola, A.; Kapur, D. Computing Uniform Interpolants for EUF via (conditional) DAG-based Compact Representations. In Proceedings of the CILC 2020, CEUR Workshop Proceedings, Rende, Italy, 13–15 October 2020; Volume 2710, pp. 67–81.
65. Ghilardi, S.; Gianola, A.; Kapur, D. Uniform Interpolants in EUF: Algorithms using DAG representations. *Log. Methods Comput. Sci.* **2022**, *under review (minor revision)*.
66. Millar, T. Model completions and omitting types. *J. Symb. Log.* **1995**, *60*, 654–672. [CrossRef]
67. Kapur, D. Shostak's Congruence Closure as Completion. In Proceedings of the RTA '97, Sitges, Spain, 2–5 June 1997; Springer: Berlin/Heidelberg, Germany, 1997; Volume 1232, pp. 23–37. [CrossRef]
68. Baader, F.; Nipkow, T. *Term Rewriting and All That*; Cambridge University Press: Cambridge, UK, 1998.
69. Calvanese, D.; Ghilardi, S.; Gianola, A.; Montali, M.; Rivkin, A. Formal Modeling and SMT-Based Parameterized Verification of Data-Aware BPMN. In Proceedings of the BPM 2019, Vienna, Austria, 1–6 September 2019; Springer: Berlin/Heidelberg, Germany, 2019; Volume 11675, pp. 157–175. [CrossRef]
70. Ghilardi, S.; Gianola, A.; Montali, M.; Rivkin, A. Petri Nets with Parameterised Data—Modelling and Verification. In Proceedings of the BPM 2020, Vienna, Austria, 1–6 September 2019; Springer: Berlin/Heidelberg, Germany, 2020; Volume 12168, pp. 55–74. [CrossRef]
71. Gianola, A. SMT-Based Safety Verification of Data-Aware Processes: Foundations and Applications. Ph.D. Thesis, Free University of Bozen-Bolzano, Bolzano, Italy, 2022.
72. Ghilardi, S.; Ranise, S. MCMT: A Model Checker Modulo Theories. In Proceedings of the IJCAR 2010, Edinburgh, UK, 16–19 July 2010; Springer: Berlin/Heidelberg, Germany, 2010; Volume 6173, pp. 22–29. [CrossRef]
73. Ghilardi, S.; Gianola, A.; Montali, M.; Rivkin, A. Delta-BPMN: A Concrete Language and Verifier for Data-Aware BPMN. In Proceedings of the BPM 2021, Rome, Italy, 6–10 September 2021; Springer: Berlin/Heidelberg, Germany, 2021; Volume 12875, pp. 179–196. [CrossRef]

74. Ghilardi, S.; Gianola, A.; Montali, M.; Rivkin, A. Petri Net-Based Object-Centric Processes with Read-Only Data. *Inf. Syst.* **2022**, *under review (minor revision)*.
75. Dumas, M.; Rosa, M.L.; Mendling, J.; Reijers, H.A. *Fundamentals of Business Process Management*, 2nd ed.; Springer: Berlin/Heidelberg, Germany, 2018. [CrossRef]