A Comprehensive Framework for Combined Decision Procedures

Silvio Ghilardi[1], Enrica Nicolini[2], and Daniele Zucchelli[1,⋆]

[1] Dipartimento di Scienze dell'Informazione
{ghilardi, zucchelli}@dsi.unimi.it
[2] Dipartimento di Matematica
nicolini@mat.unimi.it
Università degli Studi di Milano - Italy

Abstract. We define a general notion of a fragment within higher order type theory; a procedure for constraint satisfiability in combined fragments is outlined, following Nelson-Oppen schema. The procedure is in general only sound, but it becomes terminating and complete when the shared fragment enjoys suitable noetherianity conditions and allows an abstract version of a 'Keisler-Shelah like' isomorphism theorem. We show that this general decidability transfer result covers as special cases, besides applications which seem to be new, the recent extension of Nelson-Oppen procedure to non-disjoint signatures [16] and the fusion transfer of decidability of consistency of A-Boxes with respect to T-Boxes axioms in local abstract description systems [9]; in addition, it reduces decidability of modal and temporal monodic fragments [32] to their extensional and one-variable components.

1 Introduction

Decision procedures for fragments of various logics and theories play a central role in many applications of logic in computer science, for instance in formal methods and in knowledge representation. Within these application domains, relevant data appears to be heterogeneously structured, so that modularity in combining and re-using both algorithms and concrete implementations becomes crucial. This is why the development of meta-level frameworks, accepting as input specialized devices, turns out to be strategic for future advances in building powerful, fully or partially automatized systems. In this paper, we shall consider one of the most popular and simple schemata (due to Nelson-Oppen) for designing a cooperation protocol among separate reasoners; we shall plug it into a higher order framework and show how it can be used to deal with various classes of combination problems, often quite far from the originally intended application domain.

⋆ Work performed within the MIUR Project "Metodi Costruttivi in Topologia, Algebra e Fondamenti dell'Informatica". We wish to thank Manfred Schmidt-Schauss for comments on a preliminary draft of this paper.

B. Gramlich (Ed.): FroCoS 2005, LNAI 3717, pp. 1-30, 2005.

The basic feature of Nelson-Oppen method is simple: constraints involving mixed signatures are transformed into equi-satisfiable pure constraints and then the specialized reasoners try to share all the information they can acquire concerning constraints in the common subsignature, till an inconsistency is detected or till a saturation state is reached.

Nelson-Oppen method was guaranteed to be complete only for disjoint signatures and stably infinite theories, till quite recently, when it was realized [16] that stable infiniteness is just a special case of a compatibility notion, which is related to model completions of shared sub-theories. The above extension of Nelson-Oppen method to combination of theories operating over non disjoint signatures lead to various applications to decision problems in modal logics: such applications (sometimes involving non trivial extensions of the method as well as integration with other work) concerned transfer of decidability of global consequence relation to fusions [16] and to $\mathcal{E}$-connections [4,5], as well as transfer of decidability of local consequence relation to fusions [8].

Thus, most of previously existing decidability results on fusions of modal logics (for instance those in [33]) were recaptured and sometimes also improved by general automated reasoning methods based on Nelson-Oppen ideas. However, this is far from exhausting all the potentialities of such ideas and further extensions are possible. In fact, the standard approach to decision problems in modal/temporal/description logics is directly based on Kripke models (see for instance [9,15]), without the intermediation of an algebraic formalism, whereas the intermediation of the formalism of Boolean algebras with operators is essential in the approach of papers like [16,8,4,5]. The appeal to the algebraic formulation of decision problems on one side produces proofs which are much smoother and which apply also to semantically incomplete propositional logics, but on the other side it limits the method to the cases in which such a purely algebraic counterpart of semantic decision problems can be identified.

One of the main reasons for avoiding first-order formalisms in favor of propositional modal logic-style languages lies in the better computational performances of the latter. However, from a purely declarative point of view, first-order formalisms are essential in order to specify in a semantically meaningful language the relevant decision problems. This goal is mainly achieved in the case of modal logic through first-order translations, the role of such translations being simply that of codifying the intended semantics (and not necessarily that of providing computational tools).

If a semantic class $\mathcal{S}$ of Kripke frames is given, relevant decision problems are formulated as satisfiability problems (within members of $\mathcal{S}$) for standard translations of propositional modal formulae. In these formulations, unary predicates occurring in standard translations are considered *in practice* as *second order variables*: in fact, satisfiability requires the existence of suitable Kripke *models* and the latter differ from mere Kripke frames precisely by the specification of a second order assignment. The role played by second order variables becomes even more evident if we analyze the way in which standard translations of modal formulae in fusions are obtained from standard translations of formulae in the component languages. For instance, $ST(\Diamond_1\Diamond_2\mathrm{x}, w)$ is obtained by substituting into $ST(\Diamond_1\mathrm{y}, w) := \exists v(R_1(w, v) \wedge Y(v))$ the 'abstracted' second order term $\{v \mid ST(\Diamond_2\mathrm{x}, v)\} := \{v \mid \exists z(R_2(v, z) \wedge X(z))\}$ for Y (a β-conversion should follow the replacement in order to get as normal form precisely $ST(\Diamond_1\Diamond_2\mathrm{x}, w)$). Thus, even if we do not 'computationally' trust first-order logic (and consequently not even higher order logic, for much stronger reasons), it makes nevertheless sense to *analyze combination problems in the framework where they arise*, that is in the framework which is the most natural for them.

We shall work within Church's type theory: thus our syntax deals with types and terms, terms being endowed with a (codomain) type. In this higher order context, we shall provide a general definition of a fragment (more specifically of an *interpreted algebraic fragment*, see Definitions 3.2, 3.3) and of a constraint satisfiability problem, in such a way that fragments can be combined into each other and a Nelson-Oppen procedure for constraint satisfiability in combined fragments can be formally introduced.

The general procedure is only sound and specific conditions for guaranteeing termination and completeness are needed. For termination, we rely on *local finiteness* (better, on *noetherianity*) of the shared fragment, whereas for completeness we use heavy model-theoretic tools. These tools (called *isomorphism theorems*) transform equivalence with respect to satisfiability of shared atoms into isomorphism with respect to the shared signature, in such a way that satisfiability of pure constraints is not compromised. The results of this analysis is summarized in our general decidability transfer result (Theorem 5.1).

Of course, isomorphisms theorems are quite peculiar and rare. However, the classical Keisler-Shelah isomorphism theorem based on ultrapowers [11] is sufficient to justify through Theorem 5.1 the recent extension [16] of the Nelson-Oppen results to non disjoint first-order signatures and another isomorphism theorem, based on disjoint unions (better, on disjoint copies), is sufficient to justify in a similar way the decidability transfer result of [9] concerning A-Box consistency with respect to T-Boxes.[1] Having identified the conceptual core of the method, we are now able to apply it to various situations, thus getting further decidability transfer results: these results cover the combination of A-Boxes with a stably infinite first-order theory (Theorem 5.3), the combination of two so-called monadically suitable fragments (Theorem 5.6) and the combination (leading to monodic modal fragments in the sense of [32]) of a one-variable modal fragment with a monadically suitable extensional fragment (Theorem 5.7).

For space reasons, we can only explain here our settings, give examples and state the main results (for proofs and for more information, the reader is referred to the full technical report [17]).

[1] Thus, the difference between the semantically oriented proofs of [9] and the algebraically oriented proofs of papers like [16,8] seem to be mainly a question of choosing a different isomorphism theorem to justify the combined procedure.

2 Higher-Order Signatures

We adopt a type theory in Church's style (see [2,3,21] for introductions to the subject). We use letters $S_1, S_2, \dots$ to indicate *sorts* (also called *primitive types*) of a signature. Formally, sorts are a set $\mathcal{S}$ and *types over* $\mathcal{S}$ are built inductively as follows: (i) every sort $S \in \mathcal{S}$ is also a type; (ii) Ω is a type (this is called the *truth-values* type); (iii) if τ_1, τ_2 are types, so is $(\tau_1 \to \tau_2)$.

As usual external brackets are omitted; moreover, we shorten the expression $\tau_1 \to (\tau_2 \to \dots (\tau_n \to \tau))$ into $\tau_1 \dots \tau_n \to \tau$ (in this way, every type τ has the form $\tau_1 \dots \tau_n \to \tau$, where $n \geq 0$ and τ is a sort or it is Ω). In the following, we use the notation $\mathcal{T}(\mathcal{S})$ or simply $\mathcal{T}$ to indicate a *types set*, i.e. the totality of types that can be built up from the set of sorts $\mathcal{S}$. We always reserve to sorts the letters $S_1, S_2, \dots$ (as opposed to the letters τ, υ, etc. which are used for arbitrary types).

A *signature* (or a *language*) is a triple $\mathcal{L} = \langle \mathcal{T}, \Sigma, a \rangle$, where $\mathcal{T}$ is a types set, Σ is a set of constants and a is an *arity map*, namely a map $a : \Sigma \longrightarrow \mathcal{T}$. We write $f : \tau_1 \dots \tau_n \to \tau$ to express that f is a *constant of type* $\tau_1 \dots \tau_n \to \tau$, i.e. that $a(f) = \tau_1 \dots \tau_n \to \tau$. According to the above observation, we can assume that τ is a sort or that $\tau = \Omega$; in the latter case, we say that f is a *predicate* or a *relational* symbol (predicate symbols are preferably indicated with the letters $P, Q, \dots$).

We require the following *special constants* to be always present in a signature: $\top$ and $\bot$ of type Ω; $\neg$ of type $\Omega \to \Omega$; $\vee$ and $\wedge$ of type $\Omega\,\Omega \to \Omega$; $=_\tau$ of type $\tau\,\tau \to \Omega$ for each type $\tau \in \mathcal{T}$(we usually write it as '$=$' without specifying the subscript τ). The *proper symbols* of a signature are its sorts and its non special constants.

A signature is *one-sorted* iff its set of sorts is a singleton. A signature $\mathcal{L}$ is *first-order* if for any proper $f \in \Sigma$, we have that $a(f) = S_1 \dots S_n \to \tau$, where τ is a sort or it is Ω. A first-order signature is called *relational* iff any proper $f \in \Sigma$ is a relational constant, that is $a(f) = S_1 \dots S_n \to \Omega$. By contrast, a first order signature is called *functional* iff any proper $f \in \Sigma$ has arity $S_1 \dots S_n \to S$.

Given a signature $\mathcal{L} = \langle \mathcal{T}, \Sigma, a \rangle$ and a type $\tau \in \mathcal{T}$, we define the notion of an $\mathcal{L}$*-term* (or just *term*) of type τ, written $t : \tau$, as follows (for the definition we need, for every type $\tau \in \mathcal{T}$, a countable supply V_τ of variables of type τ):

- $x : \tau$ (for $x \in V_\tau$) is an $\mathcal{L}$-term of type τ;
- $c : \tau$ (for $c \in \Sigma$ and $a(c) = \tau$) is an $\mathcal{L}$-term of type τ;
- if $t : \upsilon \to \tau$ and $u : \upsilon$ are $\mathcal{L}$-terms of types $\upsilon \to \tau$ and υ, respectively, then $val_\tau(t, u) : \tau$ (also written as $t(u) : \tau$) is an $\mathcal{L}$-term of type τ;
- if $t : \tau$ is an $\mathcal{L}$-term of type τ and $x \in V_\upsilon$ is an $\mathcal{L}$-variable of type υ, $\lambda x^\upsilon\, t : \upsilon \to \tau$ is an $\mathcal{L}$-term of type $\upsilon \to \tau$.

In the following, we consider the notation x^τ (c^τ) equivalent to $x : \tau$ ($c : \tau$), where x (resp. c) is a variable (resp. a constant); if it can be deduced from the context, *the specification of the type of a term may be omitted*. Moreover, a term of type τ is also called a τ*-term* and terms of type Ω are also called *formulae*. Given a formula φ, we write $\{x \mid \varphi\}$ for $\lambda x\, \varphi$. Moreover, we shorten $val_\tau(\cdots(val_{\tau_{n-1}}(t,u_1),\cdots),u_n)$ to $t(u_1,\dots,u_n)$. Free and bound variables are defined in the usual way; we use the notation $t[x_1,\dots,x_n]$ (or $fvar(t)\subseteq\{x_1,\dots,x_n\}$) to mean that the variables occurring freely in t are included in the finite set $\underline{x}=\{x_1,\dots,x_n\}$. We often indicate finite sets or finite tuples of variables by the letters $\underline{x},\underline{y},\dots$.

Substitutions are defined in the usual way, but α-conversion (that is, bound variables renaming) might be necessary to avoid clashes. We also follow standard practice of considering terms as equivalence classes of terms under α-conversion. β- and η-conversions are defined in the standard way and we shall make use of them whenever needed (for a very brief account on the related definitions and results, the reader is referred to [12]).

For each formula φ, we define the formulae $\forall x^\upsilon\,\varphi$ and $\exists x^\upsilon\,\varphi$ as $\{x^\upsilon \mid \varphi\}=\{x^\upsilon \mid \top\}$ and as $\neg\forall x^\upsilon\neg\varphi$, respectively (the latter can also be defined differently, in an intuitionistically acceptable way, see [21]). For terms φ_1,φ_2 of type Ω, the terms $\varphi_1\to\varphi_2$ and $\varphi_1\leftrightarrow\varphi_2$ of type Ω are defined in the usual way.

By the above definitions, first-order formulae *can be considered as a subset* of the higher order formulae defined in this section. More specifically, when we speak of *first-order terms*, we mean variables $x:S$, constants $c:S$ and terms of the kind $f(t_1,\dots,t_n):S$, where $t_1,\dots,t_n$ are (inductively given) first-order terms and $a(f)=S_1\cdots S_n\to S$. Now *first-order formulae* are obtained from formulae of the kind $\top:\Omega,\bot:\Omega,P(t_1,\dots,t_n):\Omega$ (where $t_1,\dots,t_n$ are first-order terms and $a(P)=S_1\cdots S_n\to\Omega$) by applying $\exists x^S,\forall x^S,\wedge,\vee,\neg,\to,\leftrightarrow$.

In order to introduce our computational problems, we need to recall the notion of an interpretation of a type-theoretic language. Formulae of higher order type theory which are valid in ordinary set-theoretic models do not form an axiomatizable class, as it is well-known from classical limitative results. We shall nevertheless confine ourselves to standard set-theoretic models, because we are not interested in the whole type theoretic language, but only in more tractable fragments of it.

If we are given a map that assigns to every sort $S\in\mathcal{S}$ a set $[\![S]\!]$, we can inductively extend it to all types over $\mathcal{S}$, by taking $[\![\tau\to\upsilon]\!]$ to be the set of functions from $[\![\tau]\!]$ to $[\![\upsilon]\!]$ (we shall freely refer to such an extension below, without explicitly mentioning it). Given a language $\mathcal{L}=\langle\mathcal{T},\Sigma,a\rangle$, an *$\mathcal{L}$-structure* (or just a *structure*) $\mathcal{A}$ is a pair $\langle[\![-]\!]_\mathcal{A},\mathcal{I}_\mathcal{A}\rangle$, where:

(i) $[\![-]\!]_\mathcal{A}$ is a function assigning to a sort $S\in\mathcal{T}$, a (non empty, if you like) set $[\![S]\!]_\mathcal{A}$;
(ii) $\mathcal{I}_\mathcal{A}$ is a function assigning to a constant $c\in\Sigma$ of type τ, an element $\mathcal{I}_\mathcal{A}(c^\tau)\in[\![\tau]\!]_\mathcal{A}$.

In every structure $\mathcal{A}$, we require that $[\![\Omega]\!]_\mathcal{A}=\{0,1\}$ and that $\top,\bot,\neg,\wedge,\vee$ are given their standard 'truth-table' meaning.

Given an $\mathcal{L}$-structure $\mathcal{A}=\langle[\![-]\!]_\mathcal{A},\mathcal{I}_\mathcal{A}\rangle$ and a type-conformal assignment α to the variables of $\mathcal{L}$, it is possible to define (in the expected way) the interpretation $\mathcal{I}^\alpha_\mathcal{A}(t)$ of the term t under the assignment α: notice that, if t has type τ, then we have $\mathcal{I}^\alpha_\mathcal{A}(t)\in[\![\tau]\!]_\mathcal{A}$. An $\mathcal{L}$-formula φ is *satisfied* in $\mathcal{A}$ under the assignment α iff $\mathcal{I}^\alpha_\mathcal{A}(\varphi)=1$ (we usually write $\mathcal{A}\models_\alpha\varphi$ for $\mathcal{I}^\alpha_\mathcal{A}(\varphi)=1$). A formula is *satisfiable* iff it is satisfied in some structure under some assignment and a set of formulae Γ is satisfiable iff all formulas in Γ are simultaneously satisfied.

For signature inclusions $\mathcal{L}_0 \subseteq \mathcal{L}$, there is an obvious *taking reduct* operation mapping a $\mathcal{L}$-structure $\mathcal{A}$ to a $\mathcal{L}_0$-structure $\mathcal{A}_{|\mathcal{L}_0}$; we can similarly take the $\mathcal{L}_0$-reduct of an assignment, by ignoring the values assigned to variables whose types are not in $\mathcal{L}_0$ (we leave the reader to define these notions properly).

3 Fragments

General type theory is very hard to attack from a computational point of view, this is why we are basically interested only in more tractable fragments and in combinations of them. Fragments are defined as follows:

Definition 3.1. *A* fragment *is a pair* $\langle \mathcal{L}, T \rangle$ *where* $\mathcal{L} = \langle \mathcal{T}, \Sigma, a \rangle$ *is a signature and* T *is a recursive set of* $\mathcal{L}$*-terms.*

3.1 Algebraic Fragments

We want to use fragments as ingredients of larger and larger combined fragments: a crucial notion in this sense is that of an algebraic fragment.

Definition 3.2. *A fragment* $\langle \mathcal{L}, T \rangle$ *is said to be an* algebraic fragment *iff* T *satisfies the following conditions:*

(i) T is closed under composition *(that is, it is closed under substitution): if* $u[x_1, \dots, x_n] \in T$ *and* $t_i \in T$ *for all* $i = 1, \dots, n$*, then* $u[t_1, \dots, t_n] \in T$*;*
(ii) T contains domain variables*: if* τ *is a type such that some variable of type* τ *occurs free in a term* $t \in T$*, then every variable of type* τ *belongs to* T*;*
(iii) T contains codomain variables*: if* $t : \tau$ *belongs to* T*, then every variable of type* τ *belongs to* T*.*

Observe that from the above definition it follows that T is closed under renaming of terms. Quite often, one is interested in interpreting the terms of a fragment not in the class of all possible structures for the language of the fragment, but only in some selected ones (e.g. when checking satisfiability of some temporal formulae, one might be interested only in checking satisfiability in particular flows of time, those which are for instance discrete or continuous). This is the reason for 'interpreting' fragments:

Definition 3.3. *An* interpreted algebraic fragment *(to be shortened as i.a.f.) is a triple* $\Phi = \langle \mathcal{L}, T, \mathcal{S} \rangle$*, where* $\langle \mathcal{L}, T \rangle$ *is an algebraic fragment and* $\mathcal{S}$ *is a class of* $\mathcal{L}$*-structures closed under isomorphisms.*

The set of terms T in an i.a.f. $\Phi = \langle \mathcal{L}, T, \mathcal{S} \rangle$ is called the set of *Φ-terms* and the set of types τ such that $t : \tau$ is a Φ-term for some t is called the set of *Φ-types*. A *Φ-variable* is a variable x^τ such that τ is a Φ-type (or equivalently, a variable which is a Φ-term). It is also useful to *identify a (non-interpreted) algebraic fragment* $\langle \mathcal{L}, T \rangle$ with the interpreted algebraic fragment $\Phi = \langle \mathcal{L}, T, \mathcal{S} \rangle$, where $\mathcal{S}$ is taken to be the class of all $\mathcal{L}$-structures.

Definition 3.4. *Given an i.a.f. fragment Φ, a Φ-*atom *is an equation $t_1 = t_2$ between Φ-terms t_1, t_2 of the same type; a Φ-*literal *is a Φ-atom or a negation of a Φ-atom, a Φ-*constraint *is a finite conjunction of Φ-literals, a Φ-*clause *is a finite disjunction of Φ-literals. Infinite sets of Φ-literals (representing an infinite conjunction) are called* generalized Φ-constraints *(provided they contain altogether only finitely many free variables).*

Some Conventions. Without loss of generality, we may assume that $\top$ *is a Φ-atom* in every i.a.f. Φ (in fact, to be of any interest, a fragment should at least contain a term t and we can let $\top$ to be $t = t$). As a consequence, $\bot$ will always be a Φ-literal; by convention, however, we shall *include $\bot$ among Φ-atoms* (hence a Φ-atom is either an equation among Φ-terms - $\top$ included - or it is $\bot$). Since we have $\bot$ as an atom, there is no need to consider the empty clause as a clause, so clauses will be disjunctions of *at least one* literal. The reader should keep in mind these slightly non standard conventions for the whole paper.

A Φ-clause is said *positive* if only Φ-atoms occur in. A Φ-atom $t_1 = t_2$ is closed if and only if t_i is closed ($i \in \{1, 2\}$); the definition of closed Φ-literals, -constraints and -clauses is analogous. For a finite set $\underline{x}$ of variables and an i.a.f. Φ, a $\Phi(\underline{x})$-atom (-term, -literal, -clause, -constraint) is a Φ-atom (-term, -literal, -clause, -constraint) A such that $fvar(A) \subseteq \underline{x}$.

We deal in this paper mainly with the *constraint satisfiability problem* for an i.a.f. $\Phi = \langle \mathcal{L}, T, \mathcal{S} \rangle$: this is *the problem of deciding whether a Φ-constraint is satisfiable in some structure $\mathcal{A} \in \mathcal{S}$*. On the other hand, the *word problem* for Φ is the problem of deciding if the universal closure of a given Φ-atom is true in every structure $\mathcal{A} \in \mathcal{S}$.

3.2 Examples

Although there are genuinely intended higher order interpreted algebraic fragments whose word problem is decidable (see for instance Friedman theorem for simply typed λ-calculus) and also whose constraint satisfiability problem is decidable (see Rabin results on monadic second order logic), we shall mainly concentrate on examples providing applications at first-order level. The reader should however notice that we need to use higher order variables and to pay special attention to the types of a fragment in order for fragment combination defined in Subsection 4.1 to cover the desired applications.

Example 3.1 (First-order equational fragments). Let us consider a first-order language $\mathcal{L} = \langle \mathcal{T}, \Sigma, a \rangle$ (for simplicity, we also assume that $\mathcal{L}$ is one-sorted). Let T be the set of the first-order $\mathcal{L}$-terms and let $\mathcal{S}$ be an elementary class, i.e. the class of the $\mathcal{L}$-structures which happen to be the models of a certain first-order theory in the signature $\mathcal{L}$. Obviously, the triple $\Phi = \langle \mathcal{L}, T, \mathcal{S} \rangle$ is an i.a.f.. The Φ-atoms will be equalities between Φ-terms, i.e. first-order atomic formulae of the kind $t_1 = t_2$. Word problem in $\Phi = \langle \mathcal{L}, T, \mathcal{S} \rangle$ is standard uniform word problem (as defined for the case of equational theories for instance in [6]), whereas constraint satisfiability problem is the problem of deciding satisfiability of a finite set of equations and inequations.

Example 3.2 (Universal first-order fragments). The previous example disregards the relational symbols of the first-order signature $\mathcal{L}$. To take also them into consideration, it is sufficient to make some slight adjustment: besides first-order terms, also atomic formulae and $\top$, as well as propositional variables (namely variables having type Ω) will be terms of the fragment.[2] The semantic class $\mathcal{S}$ where the fragment is to be interpreted is again taken to be an elementary class. For $\Phi=\langle \mathcal{L}, T, \mathcal{S}\rangle$ so defined, the constraint satisfiability problem becomes essentially the problem of deciding the satisfiability of an arbitrary finite set of literals in the models belonging to $\mathcal{S}$.[3]

We now define different kinds of i.a.f.'s starting from the set F of first-order formulae of a first-order signature $\mathcal{L}$; for simplicity, let's suppose also that $\mathcal{L}$ is relational and one-sorted (call W this unique sort).

Example 3.3 (Full First-Order Language, plain version). We take T to be the union of F with the sets of the individual variables and of the propositional variables. Of course, $\Phi=\langle \mathcal{L}, T\rangle$ so defined is an algebraic fragment, whose types are W and Ω. By Church theorem, both word and constraint satisfiability problem are undecidable here (the two problems both reduce to satisfiability of a first-order formula with equality); they may be decidable in case the fragment is interpreted into some specific semantic class $\mathcal{S}$.

In the next example, we build formulae (out of the symbols of our fixed first order relational one-sorted signature $\mathcal{L}$) by using at most N (free or bound) individual variables; however we are allowed to use also second order variables of arity at most K:

Example 3.4 (Full First-Order Language, NK-version). Fix cardinals $K \leq N \leq \omega$ and consider, instead of F, the set F_{NK} of formulae φ that contains at most N (free or bound) individual variables and that are built up by applying boolean connectives and individual quantifiers to atomic formulae of the following two kinds:

- $P(x_{i_1}, \dots, x_{i_n})$, where P is a relational constant and $x_{i_1}, \dots, x_{i_n}$ are individual variables (since at most $x_1, \dots, x_N$ can be used, we require that $i_1, \dots, i_n \leq N$);
- $X(x_{i_1}, \dots, x_{i_n})$, where $i_1, \dots, i_n \leq N$, and X is a variable of type $W^n \to \Omega$ with $n \leq K$ (here W^n abbreviates $W \cdots W$, n-times).

The terms in the algebraic fragment $\Phi^{\mathcal{L}}_{NK} = \langle \mathcal{L}_{NK}, T^{\mathcal{L}}_{NK}\rangle$ are now the terms t such that $t \sim_{\beta\eta} \{x_1, \dots, x_n \mid \varphi\}$, for some $n \leq K$ and for some $\varphi \in F_{NK}$,

[2] Propositional variables are added here in order to fulfill Definition 3.2(iii).

[3] Notice that, by case splitting, equations $A = B$ among terms of type Ω can be replaced by $A \wedge B$ or by $\neg A \wedge \neg B$ (and similarly for inequations).

with $fvar(\varphi) \subseteq \{x_1, \dots, x_n\}$.[4] Types in such $\Phi^{\mathcal{L}}_{NK}$ are now $W^n \to \Omega$ ($n \leq K$) and this fact makes a big difference with the previous example (the difference will be sensible when combined fragments enter into the picture). Constraint satisfiability problems still reduce to satisfiability problems for sentences: in fact, once second order variables are replaced by the names of the subsets assigned to them by some assignment in an $\mathcal{L}$-structure, $\Phi^{\mathcal{L}}_{NK}$-atoms like $\{\underline{x} \mid \varphi\} = \{\underline{x} \mid \psi\}$ are equivalent to the first-order sentences $\forall \underline{x}(\varphi \leftrightarrow \psi)$ and conversely any first-order sentence θ (with at most N bound individual variables) is equivalent to the $\Phi^{\mathcal{L}}_{NK}$-atom $\theta = \top$.

The cases $N = 1, 2$ are particularly important, because in these cases the satisfiability problem for sentences (and hence also constraint satisfiability problems in our fragments) becomes decidable [22,27,24,28].

Further examples can be built by using the large information contained in the textbook [10] (see also [13]). We shall continue here by investigating fragments that arise from research in knowledge representation area, especially in connection to modal and description logics. Before, we introduce a construction that will play a central role in some applications of our results:

Definition 3.5 (Disjoint I-copy). *Consider a first order one-sorted relational signature $\mathcal{L}$ and a (non empty) index set I. The operation $\amalg_I$, defined on $\mathcal{L}$-structures and called* disjoint I-copy*, associates with an $\mathcal{L}$-structure $\mathcal{M} = \langle [\![-]\!]_{\mathcal{M}}, \mathcal{I}_{\mathcal{M}} \rangle$ the $\mathcal{L}$-structure $\amalg_I \mathcal{M}$ such that $[\![W]\!]_{\amalg_I \mathcal{M}}$ is the disjoint union of I-copies of $[\![W]\!]_{\mathcal{M}}$ (here W is the unique sort of $\mathcal{L}$). The interpretation of relational predicates is defined as follows*[5]

$$\amalg_I \mathcal{M} \models P(\langle d_1, i_1 \rangle, \dots, \langle d_n, i_n \rangle) \iff i_1 = \dots = i_n \text{ and } \mathcal{M} \models P(d_1, \dots, d_n) \tag{1}$$

for every n-ary predicate P.

Disjoint I-copy is a special case of a more general disjoint union operation: the latter is defined again by (1) and applies to any I-indexed family of structures (which may not coincide with each other).

Example 3.5 (Modal/Description Logic Fragments, global version). A *modal signature* is a set O_M, whose elements are called unary 'Diamond' modal operators.[6] O_M-modal formulae are built up from a countable set of propositional variables $\mathrm{x}, \mathrm{y}, \mathrm{z}, \dots$ by applying $\top, \bot, \neg, \wedge, \vee$ as well as the operators $\Diamond_k \in O_M$.

With every modal signature O_M we associate the first-order signature $\mathcal{L}_M$, containing a unique sort W and, for every $\Diamond_k \in O_M$, a relational constant R_k

---

4 We need to use equivalence up to $\beta\eta$-conversion here to fulfil the properties of Definition 3.2. We recall that $\beta\eta$-equivalence (noted as $\sim_{\beta\eta}$) is decided by the normalization procedure of simply typed lambda calculus.

5 Elements of the disjoint union of I-copies of a set S are represented as pairs $\langle s, i \rangle$ (meaning that $\langle s, i \rangle$ is the i-th copy of $s \in S$).

6 The case of n-ary (also non-normal) modal operators does not create special difficulties and it is left to the reader.

of type $WW \to \Omega$. Suppose we are given a bijective correspondence $\mathrm{x} \longmapsto X$ between propositional variables and second order variables of type $W \to \Omega$. Given an O_M-modal formula φ and a variable w of type W, the *standard translation* $ST(\varphi, w)$ is the $\mathcal{L}_M$-term of type Ω inductively defined as follows:

$$
\begin{array}{ll}
ST(\top, w) = \top; & ST(\bot, w) = \bot; \\
ST(\mathrm{x}, w) = X(w); & ST(\neg\psi, w) = \neg ST(\psi, w); \\
ST(\psi_1 \circ \psi_2, w) = ST(\psi_1, w) \circ ST(\psi_2, w), & \text{where } \circ \in \{\vee, \wedge\}; \\
ST(\Diamond\psi, w) = \exists v(R(w, v) \wedge ST(\psi, v)), &
\end{array}
$$

where v is a variable of type W (different from w). Let T_M be the set of those $\mathcal{L}_M$-terms t for which there exists a modal formula φ s.t. $t \sim_{\beta\eta} \{w \mid ST(\varphi, w)\}$. A *modal fragment* is an i.a.f. of the kind $\Phi_M = \langle \mathcal{L}_M, T_M, \mathcal{S}_M \rangle$, where $\mathcal{L}_M, T_M$ are as above and $\mathcal{S}_M$ is a class of $\mathcal{L}_M$-structures closed under isomorphisms and disjoint I-copies (notice that $\mathcal{L}_M$-structures, usually called *Kripke frames* in modal logic, are just sets endowed with a binary relation R_k for every $\Diamond_k \in O_M$).

Φ_M-constraints are (equivalent to) finite conjunctions of equations of the form $\{w \mid ST(\psi_i, w)\} = \{w \mid \top\}$ and of inequations of the form $\{w \mid ST(\varphi_j, w)\} \neq \{w \mid \bot\}$; such constraints are satisfied iff there exists a Kripke model[7] based on a frame in $\mathcal{S}_M$ in which the ψ_i hold globally (namely in any state), whereas the φ_j hold locally (namely in some states s_j). Thus constraint satisfiability problem becomes, in the description logics terminology, just the (simultaneous) relativized satisfiability problem for concept descriptions φ_j wrt to a given T-Box (we call *T-Box* a conjunction of Φ_M-atoms like $\{w \mid ST(\psi_i, w)\} = \{w \mid \top\}$).

Example 3.6 (Modal/Description Logic Fragments, local version). If we want to capture A-Box reasoning too, we need to build a slightly different fragment. The type-theoretic signature $\mathcal{L}_{ML}$ of our fragment is again $\mathcal{L}_M$, but T_{ML} now contains: a) the set of terms which are $\beta\eta$-equivalent to terms of the kind $ST(\varphi, w)$ (these terms are called 'concept assertions'); b) the terms of the kind $R_k(v, w)$ (these terms are called 'role assertions'); c) the variables of type W, Ω and $W \to \Omega$.

The i.a.f. $\Phi_{ML} = \langle \mathcal{L}_{ML}, T_{ML}, \mathcal{S}_{ML} \rangle$ (where $\mathcal{S}_{ML}$ is again a class of $\mathcal{L}_{ML}$-structures closed under isomorphisms and disjoint I-copies) is called an *A-Box fragment*. By a thorough case analysis [17], it is possible to show that, without loss of generality, constraints in this fragments can be represented as conjunctions of concept assertions and role assertions, plus in addition negations of role assertions and of identities among individual names. We shall call *A-Boxes* these constraints[8] and we reserve the name of *positive A-Boxes* to conjunctions of concept assertions and role assertions.

---

[7] A Kripke model is a Kripke frame together with an assignment of subsets for second order variables of type $W \to \Omega$.

[8] Standard description logics A-Boxes are just slightly more restricted, because they include only concept assertions, role assertions and also all negations of identities among distinct individual variables (by the so-called 'unique name assumption').

Example 3.7 (Modal/Description Logic Fragments, full version). If we want to deal with satisfiability of an A-Box wrt a T-Box, it is sufficient to join the two previous fragments. More precisely, we can build *full modal fragments* over a modal signature O_M, which are i.a.f.'s of the kind $\Phi_{MF} = \langle \mathcal{L}_{MF}, T_{MF}, \mathcal{S}_{MF} \rangle$, where $\mathcal{L}_{MF} = \mathcal{L}_M$, $\mathcal{S}_{MF}$ is a class of $\mathcal{L}_{ML}$-structures closed under isomorphisms and disjoint I-copies, and $T_{MF} = T_M \cup T_{ML}$. Types in these fragments are W, Ω and $W \to \Omega$; constraints are conjunctions of a T-Box and an A-Box.

Guarded and packed guarded fragments were introduced as generalizations of modal fragments [1,18,23]: in fact, they form classes of formulae which are remarkably large but still inherit relevant syntactic and semantic features of the more restricted modal formulae. In particular, guarded and packed guarded formulae are decidable for satisfiability. For simplicity, we give here the instructions on how to build only one kind of guarded fragments with equality (other similar fragments can be built by following the methods we used above).

Example 3.8 (Guarded Fragments). Let us consider a first-order one-sorted relational signature $\mathcal{L}_G$. We define the *guarded formulae* as follows:

- if $X : W \to \Omega$ and $x : W$ are variables, $X(x)$ is a guarded formula;
- if $P : W^n \to \Omega$ is a relational constant and $t_1 : W, \dots, t_n : W$ are variables, $P(t_1, \dots, t_n)$ is a guarded formula;
- if φ is a guarded formula, $\neg\varphi$ is a guarded formula;
- if φ_1 and φ_2 are guarded formulae, $\varphi_1 \wedge \varphi_2$ and $\varphi_1 \vee \varphi_2$ are guarded formulae;
- if φ is a guarded formula and π is an atomic formula such that $fvar_W(\varphi) \subseteq fvar(\pi)$ ($fvar_W(\varphi)$ are the variables of type W which occurs free in φ), then $\forall \underline{y}(\pi[\underline{x}, \underline{y}] \to \varphi[\underline{x}, \underline{y}])$ and $\exists \underline{y}(\pi[\underline{x}, \underline{y}] \wedge \varphi[\underline{x}, \underline{y}])$ are guarded formulae.

Notice that we used second order variables of type $W \to \Omega$ only (and not of type $W^n \to \Omega$ for $n > 1$): the reason, besides the applications to combined decision problems we have in mind, is that we want constraint problems to be equivalent to sentences which are still guarded, see below. Guarded formulae not containing variables of type $W \to \Omega$ are called *elementary* (or first-order) guarded formulae.

Let T_G be the set of $\mathcal{L}_G$-terms t such that t is $\beta\eta$-equivalent to a term of the kind $\{w \mid \varphi(w)\}$ (where φ is a guarded formula such that $fvar_W(\varphi(w)) \subseteq \{w\}$) and let $\mathcal{S}_G$ be a class of $\mathcal{L}_G$-structures closed under isomorphisms and disjoint I-copies: we call the i.a.f. $\Phi_G = \langle \mathcal{L}_G, T_G, \mathcal{S}_G \rangle$ a *guarded fragment*. The only type in this fragment is $W \to \Omega$ and constraint satisfiability problem in this fragment is equivalent to satisfiability of guarded sentences: this is because, in case φ_1, φ_2 are guarded formulae with $fvar_W(\varphi_i) \subseteq \{w\}$ (for $i = 1, 2$), then $\{w \mid \varphi_1\} = \{w \mid \varphi_2\}$ is equivalent to $\forall w(\varphi_1 \leftrightarrow \varphi_2)$ which is guarded (just use $w = w$ as a guard).

3.3 Reduced Fragments and Residues

If $\Phi = \langle \mathcal{L}, T, \mathcal{S} \rangle$ is an i.a.f. and $\underline{x}$ is a finite set of Φ-variables, we let $\Phi(\underline{x})$ denote the Φ-clauses whose free variables are among the $\underline{x}$. If Γ is a set of such $\Phi(\underline{x})$-clauses and $C \equiv L_1 \vee \cdots \vee L_k$ is a $\Phi(\underline{x})$-clause, we say that C is a *Φ-consequence* *of* Γ (written $\Gamma \models_\Phi C$), iff the (generalized, in case Γ is infinite) constraint $\Gamma \cup \{\neg L_1, \dots, \neg L_k\}$ is not Φ-satisfiable.

The notion of consequence is too strong for certain applications; for instance, when we simply need to delete certain deductively useless data, a weaker notion of redundancy (based e.g. on subsumption) is preferable. Our abstract axiomatization of a notion of redundancy is the following (recall that we conventionally included $\top$ and $\bot$ among Φ-atoms in any i.a.f. Φ):

Definition 3.6. *A* redundancy notion *for a fragment Φ is a recursive binary relation Red_Φ between a finite set of Φ-clauses Γ and a Φ-clause C satisfying the following properties:*

(i) $Red_\Phi(\Gamma, C)$ *implies* $\Gamma \models_\Phi C$ *(soundness);*
(ii) $Red_\Phi(\emptyset, \top)$ *and* $Red_\Phi(\{\bot\}, C)$ *both hold;*
(iii) $Red_\Phi(\Gamma, C)$ *and* $\Gamma \subseteq \Gamma'$ *imply* $Red_\Phi(\Gamma', C)$ *(monotonicity);*
(iv) $Red_\Phi(\Gamma, C)$ *and* $Red_\Phi(\Gamma \cup \{C\}, D)$ *imply* $Red_\Phi(\Gamma, D)$ *(transitivity);*
(v) if C *is subsumed by some* $C' \in \Gamma$,[9] *then* $Red_\Phi(\Gamma, C)$ *holds.*

Whenever a redundancy notion Red_Φ is fixed, we say that C is Φ-redundant wrt Γ when $Red_\Phi(\Gamma, C)$ holds.

For example, the *minimum* redundancy notion is obtained by stipulating that $Red_\Phi(\Gamma, C)$ holds precisely when ($\bot \in \Gamma$ or $C \equiv \top$ or $C \equiv \top \vee D$ or C is subsumed by some $C' \in \Gamma$). On the contrary, if the constraint solving problem for Φ is decidable, there is a maximum redundancy notion (called the *full* redundancy notion) given by the Φ-consequence relation.

Let $\Phi = \langle \mathcal{L}, T, \mathcal{S} \rangle$ be an i.a.f. on the signature $\mathcal{L} = \langle \mathcal{T}, \Sigma, a \rangle$ and let $\mathcal{L}_0 = \langle \mathcal{T}_0, \Sigma_0, a_0 \rangle$ be a subsignature of $\mathcal{L}$. The i.a.f. *restricted to* $\mathcal{L}_0$ is the i.a.f. $\Phi_{|\mathcal{L}_0} = \langle \mathcal{L}_0, T_{|\mathcal{L}_0}, \mathcal{S}_{|\mathcal{L}_0} \rangle$ that is so defined:

- $T_{|\mathcal{L}_0}$ is the set of terms obtained by intersecting T with the set of $\mathcal{L}_0$-terms;
- $\mathcal{S}_{|\mathcal{L}_0}$ consists of the structures of the kind $\mathcal{A}_{|\mathcal{L}_0}$, varying $\mathcal{A} \in \mathcal{S}$.

An i.a.f. $\Phi_0 = \langle \mathcal{L}_0, T_0, \mathcal{S}_0 \rangle$ is said to be a $\mathcal{L}_0$-*subfragment* (or simply a subfragment, leaving the subsignature $\mathcal{L}_0 \subseteq \mathcal{L}$ as understood) of $\Phi = \langle \mathcal{L}, T, \mathcal{S} \rangle$ iff $T_0 \subseteq T_{|\mathcal{L}_0}$ and $\mathcal{S}_0 \supseteq \mathcal{S}_{|\mathcal{L}_0}$. In this case, we may also say that Φ is an *expansion* of Φ_0.

Given a set Γ of $\Phi(\underline{x})$-clauses and a redundancy notion Red_{Φ_0} on a subfragment Φ_0 of Φ, we call Φ_0-*basis for* Γ a set Δ of $\Phi_0(\underline{x}_0)$-clauses such that (here $\underline{x}_0$ collects those variables among the $\underline{x}$ which happen to be Φ_0-variables):

(i) all clauses $D \in \Delta$ are positive and are such that $\Gamma \models_\Phi D$;
(ii) for every positive $\Phi_0(\underline{x}_0)$-clause C, if $\Gamma \models_\Phi C$, then C is Φ_0-redundant with respect to Δ.

Since we will be interested in exchange information concerning consequences over shared signatures, we need a notion of a residue, like in partial theory reasoning. Again, we prefer an abstract approach and treat residues as clauses which are recursively enumerated by a suitable device:

[9] As usual, this means that every literal of C' is also in C.

Definition 3.7. *Suppose we are given a subfragment Φ_0 of a fragment Φ. A* positive residue Φ-enumerator for Φ_0 *(often shortened as Φ-p.r.e.) is a recursive function mapping a finite set $\underline{x}$ of Φ-variables, a finite set Γ of $\Phi(\underline{x})$-clauses and a natural number i to a Φ_0-clause $Res^{\underline{x}}_{\Phi}(\Gamma, i)$ (to be written simply as $Res_{\Phi}(\Gamma, i)$) in such a way that:*

- *$Res_{\Phi}(\Gamma, i)$ is a positive clause;*
- *$fvar(Res_{\Phi}(\Gamma, i)) \subseteq \underline{x}$;*
- *$\Gamma \models_{\Phi} Res_{\Phi}(\Gamma, i)$ (soundness).*

*Any Φ_0-clause of the kind $Res_{\Phi}(\Gamma, i)$ (for some $i \geq 0$) will be called a Φ_0-*residue *of Γ.*

Having also a redundancy notion for Φ_0 at our disposal, we can axiomatize the notion of an 'optimized' (i.e. of a non-redundant) Φ-p.r.e. for Φ_0. The Nelson-Oppen combination procedure we give in Subsection 4.2 has *non-redundant* p.r.e.'s as main ingredients and it is designed to be 'self-adaptive' for termination in the relevant cases when termination follows from our results. These are basically the noetherian and the locally finite cases mentioned in Subsection 3.4, where p.r.e.'s which are non redundant with respect to the full redundancy notion usually exist and enjoy the termination property below.

Definition 3.8. *A Φ-p.r.e. Res_{Φ} for Φ_0 is said to be* non-redundant *(wrt a redundancy notion Red_{Φ_0}) iff it satisfies also the following properties for every $\underline{x}$, for every finite set Γ of $\Phi(\underline{x})$-clauses and for every $i \geq 0$ (we write $\Gamma_{|\Phi_0}$ for the set of clauses in Γ which are Φ_0-clauses):*

- *(i) if $Res_{\Phi}(\Gamma, i)$ is Φ_0-redundant with respect to $\Gamma_{|\Phi_0} \cup \{Res_{\Phi}(\Gamma, j) \mid j < i\}$, then $Res_{\Phi}(\Gamma, i)$ is either $\bot$ or $\top$;*
- *(ii) if $\bot$ is Φ_0-redundant with respect to $\Gamma_{|\Phi_0} \cup \{Res_{\Phi}(\Gamma, j) \mid j < i\}$, then $Res_{\Phi}(\Gamma, i)$ is equal to $\bot$;*
- *(iii) if $Res_{\Phi}(\Gamma, i)$ is equal to $\top$, then $\Gamma_{|\Phi_0} \cup \{Res_{\Phi}(\Gamma, j) \mid j < i\}$ is a Φ_0-basis for Γ.*

Definition 3.9. *A non-redundant Φ-p.r.e. for Φ_0 is said to be* complete *iff for every $\underline{x}$, for every finite set Γ of $\Phi(\underline{x})$-clauses and for every positive $\Phi_0(\underline{x})$-clause C, we have that $\Gamma \models_{\Phi} C$ implies that C is Φ_0-redundant wrt $\Gamma_{|\Phi_0} \cup \{Res_{\Phi}(\Gamma, j) \mid j \leq i\}$ for some i.*

A non-redundant Φ-p.r.e. Res_{Φ} is said to be terminating *iff for for every $\underline{x}$, for every finite set Γ of $\Phi(\underline{x})$-clauses there is an i such that $Res_{\Phi}(\Gamma, i)$ is equal to $\bot$ or to $\top$.*

Let us make a few comments on Definition 3.8: first, only non redundant residues can be produced at each step (condition (i)), if possible. If this is not possible, this means that all the relevant information has been accumulated (a Φ_0-basis has been reached). In this case, if the inconsistency $\bot$ is discovered (in the sense that it is perceived as redundant), then the residue enumeration in practice stops, because it becomes constantly equal to $\bot$ (condition (ii)).

The tautology $\top$ has the special role of marking the opposite outcome: it is the residue that is returned precisely when Γ is consistent and a Φ_0-basis has been produced, meaning that all relevant semantic consequences of Γ have been discovered (conditions (ii)-(iii)).

If the redundancy notion we use is trivial (i.e. it is the minimum one), then it is possible to show that only very mild corrections are needed for any Φ-p.r.e. for Φ_0 to become non-redundant. This observation shows that, in practice, any Φ-p.r.e. for Φ_0 can be used as input of our combined decision procedure.

3.4 Noetherian, Locally Finite and Convex Fragments

Noetherianity conditions known from Algebra say that there are no infinite ascending chains of congruences. In finitely presented algebras, congruences are represented as sets of equations among terms, hence noetherianity can be expressed there by saying that there are no infinite ascending chains of sets of atoms, modulo logical consequence. If we translate this into our general setting, we get the following notion.

An i.a.f. Φ_0 is called *noetherian* if and only if for every finite set of variables $\underline{x}$, every infinite ascending chain

$$\Theta_1 \subseteq \Theta_2 \subseteq \cdots \subseteq \Theta_n \subseteq \cdots$$

of sets of $\Phi_0(\underline{x})$-atoms is eventually constant for Φ_0-consequence (meaning that there is an n such that for all m and $A \in \Theta_m$, we have $\Theta_n \models_{\Phi_0} A$).

An i.a.f. Φ_0 is said to be *effectively locally finite* iff

(i) the set of Φ_0-types is recursive and constraint satisfiability problem for Φ_0 is decidable;
(ii) for every finite set of Φ_0-variables $\underline{x}$, there are finitely many computable $\Phi_0(\underline{x})$-terms $t_1, \dots, t_n$ such that for every further $\Phi_0(\underline{x})$-term u one of the literals $t_1 \neq u, \dots, t_n \neq u$ is not Φ_0-satisfiable (that is, in the class of the structures in which Φ_0 is interpreted, every $\Phi_0(\underline{x})$-term is equal, as an interpreted function, to one of the t_i).

The terms $t_1, \dots, t_n$ in (ii) are called the $\underline{x}$-representative terms of Φ_0.

Effective local finiteness is often used in order to make Nelson-Oppen procedures terminating [16,8,4]:[10] we shall see however that noetherianity (which is clearly a weaker condition) is already sufficient for that, once it is accompanied by a suitable effectiveness condition.

If Φ_0 is noetherian and Φ is an expansion of it, one can prove [17] that every finite set Γ of $\Phi(\underline{x})$-clauses has a finite full Φ_0-basis (i.e. there is a finite Φ_0-basis for Γ with respect to the full redundancy notion). The following noetherianity requirement for a p.r.e. is intended to be nothing but an effectiveness requirement for the computation of finite full Φ_0-bases.

[10] Notice that the above definition of local finiteness becomes slightly redundant in the first order universal case considered in these papers.

A Φ-p.r.e. Res_Φ for a noetherian fragment Φ_0 is said to be *noetherian* iff it is non redundant with respect to the full redundancy notion for Φ_0.

It is possible to prove that a noetherian Φ-p.r.e. Res_Φ for Φ_0 is terminating and also complete. Moreover, if Φ_0 is effectively locally finite and Φ is any extension of it having decidable constraint satisfiability problems, then there always exists a noetherian Φ-p.r.e. for Φ_0 [17].

Noetherianity is the essential ingredients for the termination of Nelson-Oppen combination procedures; on the other hand, for efficiency, convexity is the crucial property, as it makes the combination procedure deterministic [26]. Following an analogous notion introduced in [30], we say that an i.a.f. Φ is Φ_0-*convex* (here Φ_0 is a subfragment of Φ) iff every finite set Γ of Φ-literals having as a Φ-consequence the disjunction of $n > 1$ Φ_0-atoms, actually has as a Φ-consequence one of them.[11] Similarly, a Φ-p.r.e. for Φ_0 is Φ_0-*convex* iff $Res_\Phi(\Gamma, i)$ is always an atom (recall that by our conventions, this includes the case in which it is $\top$ or $\bot$). Any complete non-redundant Φ-p.r.e. for Φ_0 can be turned into a Φ_0-convex complete non-redundant Φ-p.r.e. for Φ_0, in case Φ is Φ_0-convex. Thus the combination procedure of Subsection 4.2 is designed in such a way that it becomes automatically *deterministic* if the component fragments are both convex with respect to the shared fragment.

An example from Algebra may help in clarifying the notions introduced in this section.

Example 3.9 (K-algebras). Given a field K, let us consider the one-sorted language $\mathcal{L}_{Kalg}$, whose signature contains the constants $0, 1$ of type V (V is the unique sort of $\mathcal{L}_{Kalg}$), the two binary function symbols $+, \circ$ of type $VV \to V$, the unary function symbol $-$ of type $V \to V$ and a K-indexed family of unary function symbols g_k of type $V \to V$. We consider the i.a.f. $\Phi_{Kalg} = \langle \mathcal{L}_{Kalg}, T_{Kalg}, \mathcal{S}_{Kalg} \rangle$ where T_{Kalg} is the set of first order terms in the above signature (we shall use infix notation for $+$ and write kv, $v_1 v_2$ for $g_k(v)$, $\circ(v_1, v_2)$, respectively). Furthermore, the class $\mathcal{S}_{Kalg}$ consists of the structures which happen to be models for the theory of (commutative, for simplicity) K-algebras: these are structures having both a commutative ring with unit and a K-vector space structure (the two structures are related by the equations $k(v_1 v_2) = (kv_1)v_2 = v_1(kv_2)$). It is clear that Φ_{Kalg} is an interpreted algebraic fragment, which is also convex and noetherian. Constraint satisfiability problem in this fragment is equivalent to the ideal membership problem and hence it is solved by Buchberger algorithm computing Gröbner bases.

As a subfragment of Φ_{Kalg} we can consider the interpreted algebraic fragment corresponding to the theory of K-vector spaces (this is also convex and noetherian, although still not locally finite). In order to obtain a noetherian Φ_{Kalg}-p.r.e. for Φ_K, we need a condition that is satisfied by common admissible term orderings, namely that membership of a linear polynomial to a finitely generated ideal to be decided only by linear reduction rules. If this happens, we

[11] When we say that a fragment Φ is *convex* tout court, we mean that it is Φ-convex. The fragments $\Phi = \langle \mathcal{L}, T, \mathcal{S} \rangle$ analyzed in Example 3.1 are convex in case $\mathcal{S}$ is the class of the models of a first-order Horn theory.

get a noetherian Φ_{Kalg}-p.r.e. for Φ_K *simply by listing the linear polynomials of a Gröbner basis.*

4 Combined Fragments

We give now the formal definition for the operation of combining fragments.

Definition 4.1. *Let $\Phi_1 = \langle \mathcal{L}_1, T_1, \mathcal{S}_1 \rangle$ and $\Phi_2 = \langle \mathcal{L}_2, T_2, \mathcal{S}_2 \rangle$ be i.a.f.'s on the languages $\mathcal{L}_1$ and $\mathcal{L}_2$ respectively; we define the* shared fragment *of Φ_1, Φ_2 as the i.a.f. $\Phi_0 = \langle \mathcal{L}_0, T_0, \mathcal{S}_0 \rangle$, where*

- $\mathcal{L}_0 := \mathcal{L}_1 \cap \mathcal{L}_2$;
- $T_0 := T_{1|\mathcal{L}_0} \cap T_{2|\mathcal{L}_0}$;
- $\mathcal{S}_0 := \mathcal{S}_{1|\mathcal{L}_0} \cup \mathcal{S}_{2|\mathcal{L}_0}$.

Thus the Φ_0-terms are the $\mathcal{L}_0$-terms that are both Φ_1-terms and Φ_2-terms, whereas the Φ_0-structures are the $\mathcal{L}_0$-structures which are reducts either of a Φ_1- or of a Φ_2-structure. According to the above definition, Φ_0 is a subfragment of both Φ_1 and Φ_2.

Definition 4.2. *The* combined fragment *of the i.a.f.'s Φ_1 and Φ_2 is the i.a.f.*

$$\Phi_1 \oplus \Phi_2 = \langle \mathcal{L}_1 \cup \mathcal{L}_2, T_1 \oplus T_2, \mathcal{S}_1 \oplus \mathcal{S}_2 \rangle$$

on the language $\mathcal{L}_1 \cup \mathcal{L}_2$ such that:

- *$T_1 \oplus T_2$ is the smallest set of $\mathcal{L}_1 \cup \mathcal{L}_2$-terms which includes $T_1 \cup T_2$, is closed under composition and contains domain and codomain variables;*
- *$\mathcal{S}_1 \oplus \mathcal{S}_2 = \{\mathcal{A} \mid \mathcal{A}$ is a $\mathcal{L}_1 \cup \mathcal{L}_2$-structure s.t. $\mathcal{A}_{|\mathcal{L}_1} \in \mathcal{S}_1$ and $\mathcal{A}_{|\mathcal{L}_2} \in \mathcal{S}_2\}$.*

$T_1 \oplus T_2$ is defined in such a way that conditions (i)-(ii)-(iii) from Definition 3.2 are matched; of course, since $\Phi_1 \oplus \Phi_2$-types turn out to be just the types which are either Φ_1- or Φ_2-types, closure under domain and codomain variables comes for free.

4.1 The Purification Steps

We say that a $\Phi_1 \oplus \Phi_2$-term is *pure* iff it is a Φ_i-term ($i = 1$ or $i = 2$) and that a $\Phi_1 \oplus \Phi_2$-constraint Γ is *pure* iff it for each literal $L \in \Gamma$ there is $i = 1$ or $i = 2$ such that L is a Φ_i-literal. Constraints in combined fragments can be purified, as we shall see.

One can effectively determine whether a given term $t \in \mathcal{L}_1 \cup \mathcal{L}_2$ belongs or not to the combined fragment: it can be shown [17] that it is sufficient to this aim to check whether it is a pure Φ_i-term and, in the negative case, to split it as $t \equiv u[t_1, \dots, t_k]$ and to recursively check whether $u, t_1, \dots, t_k$ are in the combined fragment.[12] The problem however might be computationally hard:

[12] This is well defined (by an induction on the size of t), because we do not require our terms to be in $\beta\eta$-normal form (that is, we do not require in Definition 3.2 (i) substitution to be followed by normalization).

since we basically have to guess a subtree of the position tree of the term t, *the procedure we sketched is in NP*. Notice that these complexity complications (absent in the standard Nelson-Oppen case) are due to our level of generality and that they disappear in customary situations where don't know non-determinism can be avoided by looking for 'alien' subterms, see [7] for a thorough discussion of the problem.

Let Γ be any $\Phi_1 \oplus \Phi_2$-constraint: we shall provide finite sets Γ_1, Γ_2 of Φ_1- and Φ_2-literals, respectively, such that Γ is $\Phi_1 \oplus \Phi_2$-satisfiable iff $\Gamma_1 \cup \Gamma_2$ is $\Phi_1 \oplus \Phi_2$-satisfiable. This purification process is obtained by iterated applications of the following:*

Purification Rule

$$\frac{\Gamma', A}{\Gamma',\ A([x]_P),\ x = A_{|P}} \tag{2}$$

where (we use notations like Γ', A for the constraint $\Gamma' \cup \{A\}$)

- P is a set of positions of A;
- $A_{|P}$ is a non-variable subterm of A occurring in all the positions in P (let τ be its type);
- no free variable in $A_{|P}$ is bound in A;
- x is a fresh variable of type τ;
- the literal $A([x]_P)$ (obtained by replacing in A in all the positions in P the subterm $A_{|P}$ by the variable x) is not an equation between variables;
- Γ', $A([x]_P)$, $x = A_{|P}$ is a $\Phi_1 \oplus \Phi_2$-constraint (this means that it still consists of equations and inequations among $\Phi_1 \oplus \Phi_2$-terms).

The *purification process* applies the Purification Rule as far as possible; the rule can be applied in a don't care non deterministic way (however recall that one must take care of the fact that the constraint produced by the rule still consists of $\Phi_1 \oplus \Phi_2$-literals, hence don't know non-determinism may arise inside a single application of the rule).

Proposition 4.1. *The purification process terminates and returns an equi-satisfiable constraint $\Gamma_1 \cup \Gamma_2$, where Γ_i is a set of Φ_i-literals.*

4.2 The Combination Procedure

In this subsection, we develop a procedure which is designed to solve constraint satisfiability problems in combined fragments: the procedure is sound and we shall investigate afterwards sufficient conditions for it to be terminating and complete. Let us fix relevant notation for the involved data.

* Added on July 20, (2005): The formulation of the Purification Rule has been modified according to the remarks of [17], concerning the need of the simultaneous abstraction of many occurrences of the same subterm.

Assumptions/Notational Conventions. *We suppose that we are given two i.a.f.'s* $\Phi_1 = \langle \mathcal{L}_1, T_1, \mathcal{S}_1 \rangle$ *and* $\Phi_2 = \langle \mathcal{L}_2, T_2, \mathcal{S}_2 \rangle$*, with shared fragment* $\Phi_0 = \langle \mathcal{L}_0, T_0, \mathcal{S}_0 \rangle$*. We suppose also that a redundancy notion* Red_{Φ_0} *for* Φ_0 *and two non-redundant* Φ_i*-p.r.e.'s for* Φ_0 *(call them* Res_{Φ_1}, Res_{Φ_2}*) are available.*[13] *We also fix a purified* $\Phi_1 \oplus \Phi_2$*-constraint* $\Gamma_1 \cup \Gamma_2$ *to be tested for* $\Phi_1 \oplus \Phi_2$*-consistency; we can freely suppose that* Γ_1 *and* Γ_2 *contain the same subset* Γ_0 *of* Φ_0*-literals (i.e. that* $\Gamma_0 := \Gamma_{1|\Phi_0} = \Gamma_{2|\Phi_0}$*). We indicate by* $\underline{x}_i$ *the free variables occurring in* Γ_i $(i = 1, 2)$*;* $\underline{x}_0$ *are those variables among* $\underline{x}_1 \cup \underline{x}_2$ *which happen to be* Φ_0*-variables (again we can freely suppose that* $\underline{x}_0 = \underline{x}_1 \cap \underline{x}_2$*).*

In order to describe the procedure we also need a selection function in the sense of the following definition:

Definition 4.3. *A* selection function CHOOSE(Λ) *is a recursive function accepting as input a set* Λ *of* $\Phi_0(\underline{x}_0)$*-atoms and returning a positive* $\Phi_0(\underline{x}_0)$*-clause* C *such that:*

(i) C *is a* Φ_i*-consequence of* $\Gamma_i \cup \Lambda$*, for* $i = 1$ *or* $i = 2$*;*
(ii) if $\bot$ *is* Φ_0*-redundant wrt* $\Gamma_0 \cup \Lambda$*, then* C *is* $\bot$*;*
(iii) if C *is* Φ_0*-redundant wrt* $\Gamma_0 \cup \Lambda$*, then* C *is* $\top$ *or* $\bot$*.*

The recursive function CHOOSE(Λ) will be subject also to a fairness requirement that will be explained below.

The Procedure FCOMB. *Our combined procedure generates a tree whose internal nodes are labeled by sets of* $\Phi_0(\underline{x}_0)$*-atoms; leaves are labeled by "unsatisfiable" or by "saturated". The root of the tree is labeled by the empty set and if a node is labeled by the set* Λ*, then the successors are:*

- *a single leaf labeled "unsatisfiable", if* CHOOSE(Λ) *is equal to* $\bot$*;*
- *or a single leaf labeled "saturated", if* CHOOSE(Λ) *is equal to* $\top$*;*
- *or nodes labeled by* $\Lambda \cup \{A_1\}, \dots, \Lambda \cup \{A_k\}$*, if* CHOOSE($\Lambda$) *is* $A_1 \vee \cdots \vee A_k$*.*

The branches which are infinite or end with the *"saturated"* message are called *open*, whereas the branches ending with the *"unsatisfiable"* message are called *closed*. The procedures stops (and the generation of the above tree is interrupted) iff all branches are closed or if there is an open finite branch (of course termination is not guaranteed in the general case).

Fair Selection Functions. The function CHOOSE(Λ) is *fair* iff the following happens for every open branch $\Lambda_0 \subseteq \Lambda_1 \subseteq \cdots$: if C is equal to $Res_{\Phi_i}(\Gamma_i \cup \Lambda_k, l)$ for some $i = 1, 2$ and for some $k, l \geq 0$, then C is Φ_0-redundant with respect to $\Gamma_0 \cup \Lambda_n$ for some n (roughly, *residues wrt* Φ_i *of an open branch are redundant with respect to the atoms in the branch*). Under the current assumptions/notational conventions, it can be shown that

[13] Of course, Res_{Φ_1} and Res_{Φ_2} are assumed to be both non-redundant with respect to Red_{Φ_0}.

Algorithm 1 The combination procedure

```
procedure FComb(Λ)
    C ← Choose(Λ)
    if C = ⊥ then
        return "unsatisfiable"
    else if C = ⊤ then
        return "saturated"
    end if
    for all A ∈ C do
        if FComb(Λ ∪ {A}) = "saturated" then
            return "saturated"
        end if
    end for
    return "unsatisfiable"
end procedure
```

Proposition 4.2. *There always exists a fair selection function.*

Next Proposition says that our procedure is always sound and that it terminates under noetherianity assumptions:

Proposition 4.3. *(i) If the procedure* FComb *returns "unsatisfiable", then the purified constraint $\Gamma_1 \cup \Gamma_2$ is $\Phi_1 \oplus \Phi_2$-unsatisfiable.*
(ii) If Φ_0 is noetherian and Red_{Φ_0} is the full redundancy notion, then the procedure FComb *terminates on the purified constraint $\Gamma_1 \cup \Gamma_2$.*

Completeness of the procedure FComb cannot be achieved easily, heavy conditions are needed. Since our investigations are taking a completeness-oriented route, it is quite obvious that we must consider from now on only the case in which the input Φ_i-p.r.e.'s are *complete* (see Definition 3.9). In addition we need a compactness-like assumption. We say that an i.a.f. Φ is Φ_0*-compact* (where Φ_0 is a subfragment of Φ) iff, given a Φ-constraint Γ and a generalized Φ_0-constraint Γ_0, we have that $\Gamma \cup \Gamma_0$ is Φ-satisfiable if and only if for all finite $\Delta_0 \subseteq \Gamma_0$, we have that $\Gamma \cup \Delta_0$ is Φ-satisfiable.

Since it can be shown that any extension Φ of a locally finite fragment Φ_0 is Φ_0-compact [17], if we assume effective local finiteness in order to guarantee termination, Φ_0-compactness is guaranteed too.[14]

The following Proposition gives relevant information on the semantic meaning of a run of the procedure that either does not terminate or terminates with a saturation message:

[14] Notice that only special kinds of generalized Φ-constraints are involved in the definition of Φ_0-compactness, namely those that contain finitely many proper Φ-literals; thus, Φ_0-compactness is a rather weak condition (that's why it may hold for any extension whatsoever of a given fragment, as shown by the locally finite case). Finally, it goes without saying that, by the compactness theorem for first order logic, Φ_0-compactness is guaranteed whenever Φ is a first-order fragment.

Proposition 4.4. *Suppose that Φ_1, Φ_2 are both Φ_0-compact, that the function* Choose*(Λ) is fair wrt two complete Φ_i-p.r.e.'s and that the procedure* FComb *does not return "unsatisfiable" on the purified constraint $\Gamma_1 \cup \Gamma_2$. Then there are $\mathcal{L}_i$-structures $\mathcal{M}_i \in \mathcal{S}_i$ and $\mathcal{L}_i$-assignments α_i $(i = 1, 2)$ such that:*

(i) $\mathcal{M}_1 \models_{\alpha_1} \Gamma_1$ *and* $\mathcal{M}_2 \models_{\alpha_2} \Gamma_2$*;*
(ii) for every $\Phi_0(\underline{x}_0)$-atom A, we have that $\mathcal{M}_1 \models_{\alpha_1} A$ iff $\mathcal{M}_2 \models_{\alpha_2} A$.

5 Isomorphism Theorems and Completeness

Proposition 4.4 explain what is the main problem for completeness: we would like an open branch to produce Φ_i-structures $(i = 1, 2)$ whose $\mathcal{L}_0$-reducts are isomorphic and we are only given Φ_i-structures whose $\mathcal{L}_0$-reducts are $\Phi_0(\underline{x}_0)$-equivalent (in the sense that they satisfy the same $\Phi_0(\underline{x}_0)$-atoms). Hence we need a powerful semantic device that is able to *transform $\Phi_0(\underline{x}_0)$-equivalence into $\mathcal{L}_0$-isomorphism*: this device will be called an isomorphism theorem. The precise formulation of what we mean by an isomorphism theorem needs some preparation. First of all, for the notion of an isomorphism theorem to be useful for us, it should apply to fragments extended with free constants.

Given an i.a.f. $\Phi = \langle \mathcal{L}, T, \mathcal{S} \rangle$, we denote by $\Phi(\underline{c}) = \langle \mathcal{L}(\underline{c}), T(\underline{c}), \mathcal{S}(\underline{c}) \rangle$ the following i.a.f.: (i) $\mathcal{L}(\underline{c}) := \mathcal{L} \cup \{\underline{c}\}$ is obtained by adding to $\mathcal{L}$ finitely many new constants $\underline{c}$ (the types of these new constants must be types of Φ); (ii) $T(\underline{c})$ contains the terms of the kind $t[\underline{c}/\underline{x}, \underline{y}]$ for $t[\underline{x}, \underline{y}] \in T$; (iii) $\mathcal{S}(\underline{c})$ contains precisely the $\mathcal{L}(\underline{c})$-structures whose $\mathcal{L}$-reduct is in $\mathcal{S}$. Fragments of the kind $\Phi(\underline{c})$ are called *finite expansions* of Φ.

Let $\Phi(\underline{c})$ be a finite expansion of $\Phi = \langle \mathcal{L}, T, \mathcal{S} \rangle$ and let $\mathcal{A}, \mathcal{B}$ be $\mathcal{L}(\underline{c})$-structures. We say that $\mathcal{A}$ is $\Phi(\underline{c})$-*equivalent* to $\mathcal{B}$ (written $\mathcal{A} \equiv_{\Phi(\underline{c})} \mathcal{B}$) iff for every closed $\mathcal{L}(\underline{c})$-atom A we have that $\mathcal{A} \models A$ iff $\mathcal{B} \models A$. By contrast, we say that $\mathcal{A}$ is $\Phi(\underline{c})$-*isomorphic* to $\mathcal{B}$ (written $\mathcal{A} \simeq_{\Phi(\underline{c})} \mathcal{B}$) iff there is an $\mathcal{L}(\underline{c})$-isomorphism from $\mathcal{A}$ onto $\mathcal{B}$.

We can now specify what we mean by a structural operation on an i.a.f. $\Phi_0 = \langle \mathcal{L}_0, T_0, \mathcal{S}_0 \rangle$. We will be very liberal here and define *structural operation on Φ_0* any family of correspondences $O = \{O^{\underline{c}_0}\}$ associating with any finite set of free constants $\underline{c}_0$ and with any $\mathcal{A} \in \mathcal{S}_0(\underline{c}_0)$ some $O^{\underline{c}_0}(\mathcal{A}) \in \mathcal{S}_0(\underline{c}_0)$ such that $\mathcal{A} \equiv_{\Phi_0(\underline{c}_0)} O^{\underline{c}_0}(\mathcal{A})$. If no confusion arises, we omit the indication of $\underline{c}_0$ in the notation $O^{\underline{c}_0}(\mathcal{A})$ and write it simply as $O(\mathcal{A})$.

A collection $\mathcal{O}$ of structural operations on Φ_0 *allows a Φ_0-isomorphism theorem* if and only if, for every $\underline{c}_0$, for every $\mathcal{A}, \mathcal{B} \in \mathcal{S}_0(\underline{c}_0)$, if $\mathcal{A} \equiv_{\Phi_0(\underline{c}_0)} \mathcal{B}$ then there exist $O_1, O_2 \in \mathcal{O}$ such that $O_1(\mathcal{A}) \simeq_{\Phi_0(\underline{c}_0)} O_2(\mathcal{B})$.

We shall mainly be interested into operations that can be extended to a preassigned expanded fragment. Here is the related definition. Let an i.a.f. $\Phi = \langle \mathcal{L}, T, \mathcal{S} \rangle$ extending $\Phi_0 = \langle \mathcal{L}_0, T_0, \mathcal{S}_0 \rangle$ be given; a structural operation O on Φ_0 is *Φ-extensible* if and only if for every $\underline{c}$ and every $\mathcal{A} \in \mathcal{S}(\underline{c})$ there exist $\mathcal{B} \in \mathcal{S}(\underline{c})$ such that

$$\mathcal{B}_{|\mathcal{L}_0(\underline{c}_0)} \simeq_{\Phi_0(\underline{c}_0)} O(\mathcal{A}_{|\mathcal{L}_0(\underline{c}_0)}) \quad \text{and} \quad \mathcal{B} \equiv_{\Phi(\underline{c})} \mathcal{A},$$

(where $\underline{c}_0$ denotes the set of those constants in $\underline{c}$ whose type is a Φ_0-type).

Example 5.1 (Ultrapowers). Ultrapowers [11] are basic constructions in the model theory of first-order logic. An ultrapower $\prod_{\mathcal{U}}$ (technically, an ultrafilter $\mathcal{U}$ over a set of indices is needed to describe the operation) transforms a first-order structure $\mathcal{A}$ into a first-order structure $\prod_{\mathcal{U}} \mathcal{A}$ which is elementarily equivalent to it (meaning that $\mathcal{A}$ and $\prod_{\mathcal{U}} \mathcal{A}$ satisfy the same first-order sentences). Hence if we take a fragment $\Phi_0 = \langle \mathcal{L}_0, T_0, \mathcal{S}_0 \rangle$, where $\mathcal{S}_0$ is an elementary class and $\langle \mathcal{L}_0, T_0 \rangle$ is an algebraic fragment of the kind analyzed in Example 3.3, then $\prod_{\mathcal{U}}$ is a structural operation on Φ_0. A deep result in classical model theory (known as the *Keisler-Shelah isomorphism theorem* [11]) says that two $\mathcal{L}_0$-structures $\mathcal{A}$ and $\mathcal{B}$ are elementarily equivalent iff there is an ultrafilter $\mathcal{U}$ such that the ultrapowers $\prod_{\mathcal{U}} \mathcal{A}$ and $\prod_{\mathcal{U}} \mathcal{B}$ are $\mathcal{L}_0$-isomorphic. Thus, if Φ_0 is as above, Keisler-Shelah theorem is a Φ_0-isomorphism theorem in our sense.[15] Notice also that taking the reduct of a first-order structure to a smaller signature commutes with ultrapowers, hence if $\Phi = \langle \mathcal{L}, T, \mathcal{S} \rangle$ is an extension of Φ_0 and $\mathcal{S}$ is elementary and $\langle \mathcal{L}, T \rangle$ is again a fragment of the kind analyzed in Example 3.3, then we have that the Φ_0-structural operation $\prod_{\mathcal{U}}$ is Φ-extensible (the structure $\mathcal{B}$ required in the definition of Φ-extensibility is again $\prod_{\mathcal{U}} \mathcal{A}$, where the ultrapower is now taken at the level of $\mathcal{L}$-structures).

Example 5.2 (Disjoint Copies). Consider a modal fragment Φ_{M_0} based on the empty modal signature O_{M_0} (see Example 3.5); given any non empty set I, taking disjoint I-copy $\amalg_I$ is easily seen to be a structural operation on Φ_{M_0}. Moreover, the totality of such operations (varying I) allows a Φ_{M_0}-isomorphism theorem: to show this, it is sufficient to observe that $\Phi_{M_0}(\underline{c}_0)$-equivalent structures becomes isomorphic if a sufficiently large disjoint union is applied to them, because the cardinality of subsets definable through boolean combinations of the $\underline{c}_0$'s are complete invariants for $\mathcal{L}_{M_0}(\underline{c}_0)$-isomorphism (see [17] for details).

Now notice that *a guarded elementary sentence is true in $\mathcal{M}$ iff it is true in $\amalg_I \mathcal{M}$*. Hence, taking disjoint I-copies is a Φ-extensible operation, provided Φ is a modal or a guarded fragment (in the sense of Examples 3.5 and 3.8): notice in fact that $\Phi(\underline{c})$-atoms are equivalent to elementary guarded sentences, because the second order variables of type $W \to \Omega$ have been replaced in them by the corresponding free constants $\underline{c}$ (which are constants of type $W \to \Omega$, that is they are unary first-order predicate letters).

Sometimes an isomorphism theorem does not hold precisely for a fragment $\Phi_0 = \langle \mathcal{L}_0, T_0, \mathcal{S}_0 \rangle$, but for an inessential variation (called specialization) of it. A *specialization* of Φ_0 is an i.a.f. $\Phi_0^\star$ which has the same language and the same terms as Φ_0, but whose class of $\mathcal{L}_0$-structures is a smaller class $\mathcal{S}_0^\star \subseteq \mathcal{S}_0$ satisfying the following condition: for every $\underline{c}_0$ and for every $\mathcal{A} \in \mathcal{S}_0(\underline{c}_0)$, there exists $\mathcal{A}^\star \in \mathcal{S}_0^\star(\underline{c}_0)$ such that $\mathcal{A} \equiv_{\Phi_0(\underline{c}_0)} \mathcal{A}^\star$.

[15] If $\Phi_0 = \langle \mathcal{L}_0, T_0, \mathcal{S}_0 \rangle$ is from Example 3.1-3.2 and quantifier elimination holds in $\mathcal{S}_0$, then the $\prod_{\mathcal{U}}$'s are also structural operations on Φ_0 allowing a Φ_0-isomorphism theorem (this observation is a key point for the proof of Theorem 5.2 below.)

Given an i.a.f. $\Phi = \langle \mathcal{L}, T, \mathcal{S} \rangle$ extending Φ_0, we say that Φ is *compatible* with respect to a specialization $\Phi_0^\star$ of Φ_0 if and only if for every $\underline{c}$ and $\mathcal{A} \in \mathcal{S}(\underline{c})$, there exists a $\mathcal{A}' \in \mathcal{S}(\underline{c})$ such that $\mathcal{A} \equiv_{\Phi(\underline{c})} \mathcal{A}'$ and $\mathcal{A}'_{|\mathcal{L}_0} \in \mathcal{S}_0^\star$.

Example 5.3 (Stably Infinite First-Order Theories). The Φ_0-compatibility notion is intended to recapture, in our general setting, T_0-compatibility as introduced in [16]. The latter generalizes, in its turn, the standard stable infiniteness requirement of Nelson-Oppen procedure. Let $\Phi = \langle \mathcal{L}, T, \mathcal{S} \rangle$ be an i.a.f. of the kinds considered in Example 3.1 or in Example 3.2: we say that Φ is *stably infinite* iff every satisfiable Φ-constraint is satisfiable in some infinite $\mathcal{L}$-structure $\mathcal{A} \in \mathcal{S}$.

Let now $\Phi_0 = \langle \mathcal{L}_0, T_0, \mathcal{S}_0 \rangle$ be the i.a.f. so specified: (i) $\mathcal{L}_0$ is the empty one-sorted signature; (ii) T_0 contains only the individual variables; (iii) $\mathcal{S}_0$ is the totality of $\mathcal{L}_0$-structures (i.e. the totality of sets). A specialization $\Phi_0^\star$ of Φ_0 is obtained by considering the class $\mathcal{S}_0^\star$ formed by the infinite sets.

By an easy compactness argument (compactness holds because Φ is a first-order fragment and $\mathcal{S}$ is an elementary class), it is easily seen that Φ is stably infinite iff it is compatible with respect to the specialization $\Phi_0^\star$ of Φ_0.

5.1 The Main Combination Result

By assuming the existence of Φ_i-extensible structural operations allowing a Φ_0-isomorphism theorem, it is possible to formulate a sufficient condition for our combined procedure to be complete; if we put together this condition, the termination condition of Proposition 4.3 and various remarks we made in the previous sections, we obtain the following decidability transfer result (see [17] for proof details):

Theorem 5.1. *Suppose that:*

(1) the interpreted algebraic fragments Φ_1, Φ_2 have decidable constraint satisfiability problems;
(2) the shared fragment Φ_0 is effectively locally finite (or more generally, Φ_1, Φ_2 are both Φ_0-compact, Φ_0 is noetherian and there exist noetherian positive residue Φ_1- and Φ_2-enumerators for Φ_0);
(3) Φ_1 and Φ_2 are both compatible with respect to a specialization $\Phi_0^\star$ of Φ_0;
(4) there is a collection $\mathcal{O}$ of structural operations on $\Phi_0^\star$ which are all Φ_1- and Φ_2-extensible and allow a $\Phi_0^\star$-isomorphism theorem.

Then the procedure FComb *(together with the preprocessing Purification Rule) decides constraint satisfiability in the combined fragment $\Phi_1 \oplus \Phi_2$.*

Remark. In case the shared fragment Φ_0 is locally finite, a combination procedure can be also obtained simply by guessing a maximal set Θ_0 of $\Phi_0(\underline{x}_0)$-literals and by testing the Φ_i-satisfiability of $\Theta_0 \cup \Gamma_i$. This non-deterministic version of the procedure does not require the machinery developed in Section 3.3 (but it does not apply to noetherian cases and does not yield automatic optimizations in Φ_0-convexity cases).

Remark. Theorem 5.1 cannot be used to transfer decidability of word problems to our combined fragments: the reason is that, in case the procedure FCOMB is initialized with only a single negative literal, constraints containing positive literals are nevertheless generated during the execution (and also by the Purification Rule). However, since negative literals are never run-time generated, Theorem 5.1 can be used to transfer decidability of *conditional word problems*, namely of satisfiability problems for constraints containing just one negative literal.

5.2 Applications: Decidability Transfer through Ultrapowers

We shall use the Keisler-Shelah isomorphism Theorem of Example 5.1 to get the transfer decidability result of [16] as a special case of Theorem 5.1.

Let $\Phi_1 = \langle \mathcal{L}_1, T_1, \mathcal{S}_1 \rangle$ and $\Phi_2 = \langle \mathcal{L}_2, T_2, \mathcal{S}_2 \rangle$ be i.a.f.'s of the kinds considered in the Example 3.1 or in Example 3.2 and let $\Phi_0 = \langle \mathcal{L}_0, T_0, \mathcal{S}_0 \rangle$ be their shared fragment. The hypothesis for the decidability transfer result of [16] are the following:

(C1) there is a universal theory T_0 in the shared signature $\mathcal{L}_0$ such that every $\mathcal{A} \in \mathcal{S}_0$ is a model of T_0;
(C2) T_0 admits a model-completion $T_0^\star$;[16]
(C3) for $i = 1, 2$, every $\mathcal{A} \in \mathcal{S}_i$ embeds into some $\mathcal{A}' \in \mathcal{S}_i$ which is a model of $T_0^\star$;
(C4) Φ_0 is effectively locally finite.

Theorem 5.2 ([16]). *Suppose that Φ_1 and Φ_2 are i.a.f.'s of the kinds considered in Examples 3.1-3.2, which moreover satisfy conditions* (C1)-(C4) *above. If constraint satisfiability problems are decidable in Φ_1 and Φ_2, then they are decidable in $\Phi_1 \oplus \Phi_2$ too.*

If we take as T_0 the empty theory (in the one-sorted first-order empty language with equality), then $T_0^\star$ is the theory of an infinite set and condition (C3) is equivalent to stable infiniteness (by a simple argument based on compactness); thus, Theorem 5.2 *reduces to the standard Nelson-Oppen result* [25,26,31] *concerning stably infinite theories over disjoint signatures.* We recall from [16] that among relevant examples of theories to which Theorem 5.2 is easily seen to apply, we have Boolean algebras with operators (namely the theories axiomatizing algebraic semantics of modal logic): thus, decidability of conditional word problem transfers from two theories axiomatizing varieties of modal algebras with operators to their union (provided only Boolean operators are shared). This result, proved in [33] by specific techniques, is the algebraic version of the *fusion transfer of decidability of global consequence relation in modal logic.*

We remark that condition (C4) can be weakened to

(C4′) Φ_0 is noetherian and there exist noetherian positive residue Φ_1- and Φ_2-enumerators for Φ_0,

[16] We refer the reader to [16] for the definition and to any textbook on model theory like [11] for more information.

as suggested by Theorem 5.1 (2). As an example of an application of Theorem 5.2 under this weaker condition one can consider the theory of K-algebras endowed with a linear endomorphism: this theory is the combination of the theory of K-algebras and of the theory of K-vector spaces endowed with an endomorphism (positive residue enumerators for the noetherian shared fragment can be obtained in both cases by the method outlined in Example 3.9).

As another application of Theorem 5.1 based on Keisler-Shelah isomorphism theorem, we show *how to include a first order equational theory within description logic A-Boxes*. To get a decidability transfer result for the combination of an equational i.a.f. $\Phi = \langle \mathcal{L}, T, \mathcal{S} \rangle$ from Example 3.1 and of an A-Box fragment $\Phi_{ML} = \langle \mathcal{L}_{ML}, T_{ML}, \mathcal{S}_{ML} \rangle$ from Example 3.6, we only need mild additional hypotheses. These are explained in the statement of the following Theorem:

Theorem 5.3. *Suppose that we are given an equational i.a.f. $\Phi = \langle \mathcal{L}, T, \mathcal{S} \rangle$ from Example 3.1 and an A-Box fragment $\Phi_{ML} = \langle \mathcal{L}_{ML}, T_{ML}, \mathcal{S}_{ML} \rangle$ from Example 3.6; suppose also that the signatures $\mathcal{L}$ and $\mathcal{L}_{ML}$ are disjoint, that Φ is stably infinite and that $\mathcal{S}_{ML}$ is an elementary class. Then decidability of constraint satisfiability problems transfers from Φ and Φ_{ML} to $\Phi \oplus \Phi_{ML}$.*

Notice that the fragment $\Phi \oplus \Phi_{ML}$ of Theorem 5.3 is quite peculiar (combined terms all arise from a single composition step).

5.3 Applications: Decidability Transfer through Disjoint Copies

Disjoint copies are the key tool for transfer decidability results in modal fragments. If O_{M_1} and O_{M_2} are modal signatures, we let $O_{M_1 \oplus M_2}$ indicate their disjoint union ($O_{M_1 \oplus M_2}$ is called the fusion of the modal signatures O_{M_1} and O_{M_2}). Given a modal i.a.f. Φ_{M_1} over O_{M_1} and a modal i.a.f. Φ_{M_2} over O_{M_2} (see Example 3.5), let us define their *fusion* as the modal i.a.f.

$$\Phi_{M_1 \oplus M_2} = \langle \mathcal{L}_{M_1 \oplus M_2}, T_{M_1 \oplus M_2}, \mathcal{S}_{M_1} \oplus \mathcal{S}_{M_2} \rangle .$$

Theorem 5.1 and the considerations in Example 5.2 show that decidability of constraint satisfiability transfers from two modal i.a.f.'s Φ_{M_1} and Φ_{M_2} (operating on disjoint modal signatures) to their combination $\Phi_{M_1} \oplus \Phi_{M_2}$. Since it can be shown that the latter differs from the fusion $\Phi_{M_1 \oplus M_2}$ only by trivial $\beta\eta$-conversions, the following well-known decidability transfer result obtains:

Theorem 5.4 (Decidability transfer for modal i.a.f.'s). *If two modal interpreted algebraic fragments Φ_{M_1} and Φ_{M_2} have decidable constraint satisfiability problems, so does their fusion $\Phi_{M_1 \oplus M_2}$.*

Fragments of the kind examined in Example 3.6 are not interesting for being combined with each other, because the absence of the type $W \to \Omega$ makes such combinations trivial. On the contrary, full modal fragments from Example 3.7 are quite interesting in this respect (we recall that they reproduce both A-Box and T-Box reasoning from the point of view of description logics). Under the obvious definition of fusion for full modal i.a.f.'s, we have the following result (the proof requires just slight modifications to the considerations of Example 5.2):

Theorem 5.5 (Decidability transfer for full modal i.a.f.'s). *If two full modal i.a.f.'s have decidable constraint satisfiability problems, so does their fusion.*

Theorem 5.5 (once completed with the straightforward extension to n-ary non normal modalities) covers the results of [9] on transfer of decidability of A-Box consistency (wrt T-Boxes axioms) in fusions of local abstract description systems.

We now try to extend our decidability transfer results to appropriate combinations of guarded or of two-variable fragments. However, to get positive results, we need to keep shared signatures under control (otherwise undecidability phenomena arise). In addition, we still want to exploit the isomorphism theorem of Example 5.2 and for that we need the shared signature to be empty and second order variables appearing as terms in the fragments to be monadic only. The kind of combination that arise in this way is a form of fusion, that we shall call monadic fusion. We begin by identifying a class of fragments to which our techniques apply.

Let us call $\Phi_\emptyset = \langle \mathcal{L}_\emptyset, T_\emptyset, \mathcal{S}_\emptyset \rangle$ the following i.a.f.: (i) $\mathcal{L}_\emptyset$ is the empty one-sorted first-order signature (that is, $\mathcal{L}_\emptyset$ does not contain any proper symbol, except for its unique sort which is called D); (ii) $T_\emptyset$ is equal to $T_{11}^{\mathcal{L}_\emptyset}$;[17] (iii) $\mathcal{S}_\emptyset$ contains all $\mathcal{L}_\emptyset$-structures.

Definition 5.1. *A* monadically suitable[18] *i.a.f.* $\Phi = \langle \mathcal{L}, T, \mathcal{S} \rangle$ *is an i.a.f. such that:*

(i) $\mathcal{L}$ *is a relational one-sorted first-order signature;*
(ii) $T_{11}^{\mathcal{L}_\emptyset} \subseteq T \subseteq T_{\omega 1}^{\mathcal{L}}$*;*
(iii) the $\Phi_\emptyset$*-structural operation of taking disjoint* I*-copies is* Φ*-extensible.*

As a first example of a monadically suitable fragment, we can consider the guarded fragments of Example 3.8 (see also the considerations in Example 5.2). To get another family of examples, we introduce an alternative construction for proving extensibility of the operation of taking disjoint I-copies. This construction is nicely behaved only for fragments without identity and is called I-conglomeration:

Definition 5.2 (I-conglomeration). *Consider a first order one-sorted relational signature* $\mathcal{L}$ *and a (non empty) index set* I*. The operation* $\amalg^I$*, defined*

[17] See Example 3.4 for this notation and for other similar notation used below.

[18] We remark that, despite the fact that the definition of a monadically suitable fragment needs the present paper settings to be formulated, there is some anticipation of it in the literature on monodic fragments (see for instance statements like that of Theorem 11.21 in [15]).

on $\mathcal{L}$-structures and called I-conglomeration*, associates with an $\mathcal{L}$-structure $\mathcal{M} = \langle [\![-]\!]_{\mathcal{M}}, \mathcal{I}_{\mathcal{M}} \rangle$ the $\mathcal{L}$-structure $\amalg^I \mathcal{M}$ such that $[\![D]\!]_{\amalg^I \mathcal{M}}$ is the disjoint union of I-copies of $[\![D]\!]_{\mathcal{M}}$ (here D is the unique sort of $\mathcal{L}$). The interpretation of relational constants is defined in the following way*

$$\amalg^I \mathcal{M} \models P(\langle d_1, i_1 \rangle, \ldots, \langle d_n, i_n \rangle) \quad \Longleftrightarrow \quad \mathcal{M} \models P(d_1, \ldots, d_n)$$

for every n-ary relational predicate P different from equality.

Notice that I-conglomerations and disjoint I-copies *coincide* for relational first order signatures having only unary predicates.

Example 5.4. Let $\mathcal{L}_{2V}$ be a first-order relational one-sorted signature; a *two variables i.a.f.* over $\mathcal{L}_{2V}$ is a fragment of the kind $\Phi_{2V} = \langle \mathcal{L}_{2V}, T_{2V}, \mathcal{S}_{2V} \rangle$, where: (i) T_{2V} contains the terms *without identity* which belongs to the set $T_{NK}^{\mathcal{L}_{2V}}$ of Example 3.4 for $K = 1$ and $N = 2$; (ii) $\mathcal{S}_{2V}$ is a class of $\mathcal{L}_{2V}$-structures closed under isomorphisms and I-conglomerations. To show that Definition 5.1 applies to Φ_{2V}, it is sufficient to check that a first order formula not containing the equality predicate is satisfiable in $\mathcal{M}$ iff it is satisfiable in $\amalg^I \mathcal{M}$.

For two monadically suitable i.a.f.'s Φ_1 and Φ_2 operating on disjoint signatures, let us call the combined fragment $\Phi_1 \oplus \Phi_2$ the *monadic fusion* of Φ_1 and Φ_2. For monadic fusions we have the following [17]:

Theorem 5.6 (Decidability transfer for monadically suitable i.a.f.'s). *If two monadically suitable i.a.f.'s Φ_1, Φ_2 operating on disjoint signatures have decidable constraint satisfiability problems, so does their monadic fusion.*

Theorem 5.6 offers various combination possibilities, however notice that: (a) the conditions for a fragment to be monadically suitable are rather strong (for instance, the two variable fragment with identity is not monadically suitable); (b) the notion of monadic fusion is a restricted form of combination, because only unary second order variables are available for replacement when forming formulae of the combined fragment.

5.4 Applications: Decidability Transfer for Monodic Fragments

Fragments in first-order *modal* predicate logic become undecidable quite soon: for instance, classical decidability results for the monadic or the two-variables cases do not extend to modal languages [20,14,19]. However there still are interesting modal predicate fragments which are decidable: one-variable fragments are usually decidable [29,15], as well as many monodic fragments. We recall that a *monodic* formula is a modal first order formula in which modal operators are applied only to subformulae containing at most one free variable. Monodic fragments whose extensional (i.e. non modal) component is decidable seem to be decidable too [32,15]: we shall give this fact a formulation in terms of a decidability transfer result for monodic fragments which are obtained as combinations of a suitable extensional fragment and of a one-variable first-order modal fragment. Since we prefer, for simplicity, not to introduce a specific formal notion of a modal fragment, we shall proceed through standard translations and rely on our usual notion of an i.a.f..

Constant Domains and Standard Translation. *Modal predicate formulae* are built up from atomic formulae of a given first-order one-sorted relational signature $\mathcal{L}$ and from formulae of the kind $X(x)$ (where X is a unary second order variable), by using boolean connectives, individual quantifiers and a diamond operator $\Diamond$.[19]

There are actually different standard translations for first-order modal languages, we shall concentrate here on the translation corresponding to *constant domain* semantics. The latter is defined as follows. The signature $\mathcal{L}^W$ has, in addition to the unique sort D of $\mathcal{L}$, a new sort W; relational constants of type $D^n \to \Omega$ have corresponding relational constants in $\mathcal{L}^W$ of type $D^n W \to \Omega$. We use equal names for corresponding constants: this means for instance that if P has type $D^2 \to \Omega$ in $\mathcal{L}$, the same P has type $D^2 W \to \Omega$ in $\mathcal{L}^W$. We shall make the same conventions for second order variables: hence a second order $\mathcal{L}$-variable X of type $D \to \Omega$ has a corresponding second order variable X of type $DW \to \Omega$ in $\mathcal{L}^W$.

Notice that a $\mathcal{L}^W$-structure $\mathcal{A}$ is nothing but a $[\![W]\!]_{\mathcal{A}}$-indexed class of $\mathcal{L}$-structures, all having the same domain $[\![D]\!]_{\mathcal{A}}$: we indicate by $\mathcal{A}_w$ the structure corresponding to $w \in [\![W]\!]_{\mathcal{A}}$ and call it the *fiber structure over* w. The signature $\mathcal{L}^{WR}$ is obtained from $\mathcal{L}^W$ by adding it also a binary 'accessibility' relation R of type $WW \to \Omega$. This is the signature we need for defining the standard translation.

For a modal predicate $\mathcal{L}$-formula $\varphi[x_1^D, \dots, x_n^D]$ and for a variable $w : W$, we define the (non modal) $\mathcal{L}^{WR}$-formula $ST(\varphi, w)$ as follows:

$$\begin{array}{ll} ST(\top, w) = \top; & ST(\bot, w) = \bot; \\ ST(P(x_{i_1}, \dots, x_{i_m}), w) = P(x_{i_1}, \dots, x_{i_m}, w); & ST(X(x_i), w) = X(x_i, w); \\ ST(\neg\psi, w) = \neg ST(\psi, w); & ST(\exists x^D \psi, w) = \exists x^D ST(\psi, w); \\ ST(\psi_1 \circ \psi_2, w) = ST(\psi_1, w) \circ ST(\psi_2, w), & \text{where } \circ \in \{\vee, \wedge\}; \\ ST(\Diamond\psi, w) = \exists v^W (R(w, v) \wedge ST(\psi, v)). & \end{array}$$

Monodic Fusions for Fragments Let $\mathcal{F}_{1M}$ be a class of Kripke frames closed under disjoint unions and isomorphisms. We call *one-variable modal fragment* induced by $\mathcal{F}_{1M}$ the i.a.f. $\Phi_{1M} = \langle \mathcal{L}_{1M}, T_{1M}, \mathcal{S}_{1M} \rangle$, where: (i) $\mathcal{L}_{1M} := \mathcal{L}_{\emptyset}^{WR}$, where $\mathcal{L}_{\emptyset}$ is the empty one-sorted first-order signature;(ii) T_{1M} contains the terms which are $\beta\eta$-equivalent to terms of the kind $\{w^W, x^D \mid ST(\varphi, w)\}$, where φ is a modal predicate formula having x as the only (free or bound) variable;

[19] All the results in this subsection extend to the case of multimodal languages and to the case of n-ary modalities like Since, Until, etc.

(iii) $\mathcal{S}_{1M}$ is the class of the $\mathcal{L}_{1M}$-structures $\mathcal{A}$ such that $[\![D]\!]_{\mathcal{A}}$ is not empty and such that the Kripke frame $([\![W]\!]_{\mathcal{A}}, \mathcal{I}_{\mathcal{A}}(R))$ belongs to $\mathcal{F}_{1M}$.

For a *monadically suitable* i.a.f. $\Phi_e = \langle \mathcal{L}_e, T_e, \mathcal{S}_e \rangle$ (recall Definition 5.1), we define the i.a.f. $\Phi_e^W = \langle \mathcal{L}_e^W, T_e^W, \mathcal{S}_e^W \rangle$, as follows: (i) T_e^W contains the terms of the kind $\{w^W, x^D \mid ST(\varphi, w)\}$, for $\{x^D \mid \varphi\} \in T_e$: (ii) $\mathcal{S}_e^W$ contains the $\mathcal{L}_e^W$-structures $\mathcal{A}$ whose fibers $\mathcal{A}_w$ are all in $\mathcal{S}_e$.

Fix a one variable modal fragment Φ_{1M} and a first-order monadically suitable fragment Φ_e; we call *monodic fusion* of Φ_e and Φ_{1M} the combined fragment $\Phi_e^W \oplus \Phi_{1M}$.

Thus one may for instance combine guarded or two-variables fragments[20] with one-variables modal fragments to get monodic fusions corresponding to the relevant cases analyzed in [32,15]. In fact (modulo taking standard translation), in combined fragments like $\Phi_e^W \oplus \Phi_{1M}$ we can begin with formulae $\varphi[x]$ of Φ_e, apply to them a modal operator, then use the formulae so obtained to replace second order variables in other formulae from Φ_e, etc. Fragments of the kind $\Phi_e^W \oplus \Phi_{1M}$ formalize the intuitive notion of a monodic modal fragment whose extensional component is Φ_e. Since Φ_{1M} is also interpreted, constraint satisfiability in $\Phi_e^W \oplus \Phi_{1M}$ is restricted to a desired specific class of modal frames/flows of time.

Theorem 5.7. *If the one variable modal i.a.f. Φ_{1M} and the monadically suitable i.a.f. Φ_e have decidable constraint satisfiability problems, then their monodic fusion $\Phi_e^W \oplus \Phi_{1M}$ also has decidable constraint satisfiability problems.*

The proof of Theorem 5.7 reduces the statement to be proved to Theorem 5.1, after translating our fragments into fragments of a language describing appropriate *descent data* [17] (disjoint I-copies and fiberwise disjoint I-copies then provide the suitable isomorphism theorem).

6 Conclusions

In this paper we introduced a type-theoretic machinery in order to deal with the combination of decision problems of various nature. Higher order type theory has been essentially used as a *unifying specification language*; we have also seen how the *types interplay* can be used in a rather subtle way to design combined fragments and consequently appropriate constraints satisfiability problems.

Decision problems are at the heart of logic and of its applications, that's why they are so complex and irregularly behaved. Given that it is very difficult (and presumably impossible) to get satisfying general results in this area, the emphasis should concentrate on *methodologies* which are capable of solving entiere classes of concrete problems. Among methodologies, we can certainly include *methodologies for combination*: these may be very helpful when the solution of a problem can be modularly decomposed or when the problem itself appears to be heterogeneous in its nature.

[20] We recall that two-variable fragments are monadically suitable only if we take out identity.

In this paper, we took into consideration *Nelson-Oppen methodology* (which is probably the simplest combination methodology) and tried to push it as far as possible. Surprisingly, it turned out that it might be quite powerful, when *joined to strong model theoretic results* (the isomorphism theorems). Thus, we tried to give the reader a gallery of different applications that can be solved in a *uniform way* by this methodology. Some of these applications are new, some other summarize recent work by various people. New problems certainly arise now: they concern both further applications of Nelson-Oppen schema and the individuation or more sophisticated schemata, for the problems that cannot be covered by the Nelson-Oppen approach. We hope that the higher order framework and the model theoretic techniques we introduced in this paper may give further contributions within this research perspective.

References

1. H. Andréka, I. Nemeti, and J. VanBenthem. Modal languages and bounded fragments of predicate logics. *Journal of Philosophical Logic*, 27:217–274, 1998.
2. Peter B. Andrews. *An introduction to mathematical logic and type theory: to truth through proof*, volume 27 of *Applied Logic Series*. Kluwer Acad. Publ., 2002.
3. Peter B. Andrews. Classical type theory. In A. Robinson and A. Voronkov, editors, *Handbook of Automated Reasoning*, volume II, pages 966–1007. Elsevier/MIT, 2001.
4. Franz Baader and Silvio Ghilardi. Connecting many-sorted theories. In *Proceedings of the 20th International Conference on Automated Deduction (CADE-05)*, Lecture Notes in Artificial Intelligence, 2005.
5. Franz Baader and Silvio Ghilardi. Connecting many-sorted structures and theories through adjoint functions. In *Proceedings of the 5th International Workshop on Frontiers of Combining Systems (FROCOS-05)*, Lecture Notes in Artificial Intelligence, 2005.
6. Franz Baader and Tobias Nipkow. *Term rewriting and all that.* Cambridge University Press, Cambridge, 1998.
7. Franz Baader and Cesare Tinelli. Deciding the word problem in the union of equational theories. *Information and Computation*, 178(2):346–390, December 2002.
8. Franz Baader, Silvio Ghilardi, and Cesare Tinelli. A new combination procedure for the word problem that generalizes fusion decidability results in modal logics. *Information and Computation.* (to appear).
9. Franz Baader, Carsten Lutz, Holger Sturm, and Frank Wolter. Fusions of description logics and abstract description systems. *Journal of Artificial Intelligence Research*, 16:1–58, 2002.
10. Egon Börger, Erich Grädel, and Yuri Gurevich. *The classical decision problem.* Universitext. Springer-Verlag, Berlin, 2001.
11. Chen-Chung Chang and H. Jerome Keisler. *Model Theory.* North-Holland, Amsterdam-London, IIIrd edition, 1990.
12. G. Dowek. Higher order unification and matching. In A. Robinson and A. Voronkov, editors, *Handbook of Automated Reasoning*, volume II, pages 1009–1062. Elsevier/MIT, 2001.
13. C. Fermüller, A. Leitsch, T. Tammet, and N. Zamov. *Resolution methods for the decision problem*, volume 679 of *Lecture Notes in Computer Science.* Springer-Verlag, 1993. Lecture Notes in Artificial Intelligence.

14. D. M. Gabbay and V.B. Shehtman. Undecidability of modal and intermediate first-order logics with two individual variables. *Journal of Symbolic Logic*, 58:800–823, 1993.
15. D.M. Gabbay, A. Kurucz, F. Wolter, and M. Zakharyaschev. *Many-Dimensional Modal Logics: Theory and Applications*, volume 148 of *Studies in Logic and the Foundations of Mathematics*. Elsevier, 2003.
16. Silvio Ghilardi. Model theoretic methods in combined constraint satisfiability. *Journal of Automated Reasoning*, 33(3-3):221–249, 2005.
17. Silvio Ghilardi, Enrica Nicolini, and Daniele Zucchelli. A comprehensive framework for combined decision procedures. Technical Report DSI 304-05, Università degli Studi di Milano, 2005. URL `http://homes.dsi.unimi.it/~zucchell/publications/techreport/GhiNiZu-RI304-05.pdf`.
18. E. Grädel. Decision procedures for guarded logics. In *Proceedings of CADE-16*, volume 1632 of *Lecture Notes in Computer Science*, pages 31–51. Springer, 1999.
19. R. Kontchakov, A. Kurucz, and M. Zakharyaschev. Undecidability of first-order intuituionistic and modal logics with two variables. 2004. manuscript.
20. S. Kripke. The undecidability of monadic modal quantificational theory. *Z. Math. Logik Grundlag. Math.*, 8:113–116, 1962.
21. J. Lambek and P. J. Scott. *Introduction to higher order categorical logic*, volume 7 of *Cambridge Studies in Advanced Mathematics*.
22. L. Löwhenheim. Über Möglichkeiten im Relativkalkül. *Math. Annalen*, 76:228–251, 1915.
23. M. Marx. Tolerance logic. *Journal of Logic, Language and Information*, 10:353–374, 2001.
24. M. Mortimer. On languages with two variables. *Z. Math. Logik Grundlag. Math.*, 21:135–140, 1975.
25. Greg Nelson and Derek C. Oppen. Simplification by cooperating decision procedures. *ACM Trans. on Programming Languages and Systems*, 1(2):245–257, October 1979.
26. Derek C. Oppen. Complexity, convexity and combinations of theories. *Theoretical Computer Science*, 12:291–302, 1980.
27. D. Scott. A decision method for for validity of sentences in two variables. *Journal of Symbolic Logic*, 27:477, 1962.
28. K. Segerberg. Two-dimensional modal logic. *Journal of Philosophical Logic*, 2: 77–96, 1973.
29. V.B. Shehtman. On some two-dimensional modal logics. In *8th Congress on Logic Methodology and Philosophy of Science, vol. 1*, pages 326–330. Nauka, Moskow, 1987.
30. Cesare Tinelli. Cooperation of background reasoners in theory reasoning by residue sharing. *Journal of Automated Reasoning*, 30(1):1–31, 2003.
31. Cesare Tinelli and Mehdi T. Harandi. A new correctness proof of the Nelson–Oppen combination procedure. In *Proc. of the 1st Int. Workshop on Frontiers of Combining Systems*, pages 103–120. Kluwer Acad. Publ., 1996.
32. F. Wolter and M. Zakharyaschev. Decidable fragments of first-order modal logics. *Journal of Symbolic Logic*, 66:1415–1438, 2001.
33. Frank Wolter. Fusions of modal logics revisited. In M. Kracht, M. de Rijke, H. Wansing, and M. Zakharyaschev, editors, *Advances in Modal Logic*. CSLI, Stanford, CA, 1998.