Combining Equational Theories Sharing Non-Collapse-Free Constructors

Franz Baader[1*] and Cesare Tinelli[2**]

[1] LuFg Theoretical Computer Science, RWTH Aachen.
baader@informatik.rwth-aachen.de
[2] Department of Computer Science, University of Iowa.
tinelli@cs.uiowa.edu

Abstract. In this paper we extend the applicability of our combination method for decision procedures for the word problem to theories sharing *non-collapse-free* constructors. This extension broadens the scope of the combination procedure considerably, for example in the direction of equational theories axiomatizing the equivalence of modal formulae.

1 Introduction

The word problem for a theory E is concerned with the question of whether two terms are equal in all models of E. In [4] we provided modular decidability results for the word problem in the case of unions of equational theories with possibly non-disjoint signatures, subsuming previous well-known results on the decidability of the word problem for the union of equational theories with disjoint signatures [12, 11, 13]. Our results were achieved by assuming that the function symbols shared by the component theories were *constructors* in a appropriate sense.

The notion of constructors presented in [4] was modeled after one first introduced in [14], and generalized that in [5]. Its formulation is based on the observation that some equational theories are such that the reducts of their free models to a subset Σ of their signature are themselves free. We would call constructors the symbols in Σ. The actual definition in [4], however, incorporated the restriction that the equational theory of the constructors had to be collapse-free.[1] This restriction was essentially technical, as it was used to provide a syntactic characterizations of the generators of the free Σ-reducts in terms of a certain set G of terms, which was then utilized in various proofs in the paper.

In the present paper, by using a more general way of defining the set G above, we remove the collapse-freeness restriction and show that all the combination results given in [4] continue to hold without it.

* Partially supported by the EC Working Group CCL II.
** Partially supported by the National Science Foundation grant no. CCR-99-72311.

[1] In other words, no non-variable term over the constructor symbols could be equivalent in E to one of its variables.

In [4] we used a rule-based procedure for combining in a modular way a procedure deciding the word problem for a theory E_1 and a procedure deciding the word problem for a theory E_2 into a procedure deciding the word problem for the theory $E_1 \cup E_2$. As mentioned, the main requirement was that the symbols shared by E_1 and E_2 were constructors for each of them. In this paper, we obtain the generalized combination results by using a proper modification of the procedure, which does not rely anymore on the assumption that the constructor theory is collapse-free.

The net effect of lifting the collapse-freeness restriction is a considerable expansion of the scope of our combination results. A lot more equational theories obtained as a conservative extension of a core Σ-theory are now such that Σ is a set of constructors for them. Which means, potentially, that a lot more theories built as conservative extensions of a same Σ-theory can be combined with our method.[2]

One particularly interesting class of such theories includes the equational axiomatizations of some (propositional) modal logics, on which we give more details in Sect. 2.2. A fair amount of research has been done on the combination of modal logics. We believe that our results for the word problems can now be used to contribute to this research by recasting the combination of two modal logics as the union of their corresponding equational theories. However, we have not yet had the time to explore these possibilities in more depth. We are working on this in a joint project with modal logicians.

For now, we present and discuss our generalized notion of constructors, and provide some examples of theories admitting constructors in the new sense but not in the old one, including an equational theory corresponding to a modal logic. Then, we describe the modified version of the combination procedure, provide a sketch of its correctness proof, and show how that leads to exactly the same results given in [4], but of course with the wider scope provided by the new definition of constructors. Because of space limitations, we refer the reader to the longer version of this paper [3] for detailed proofs of our results.

2 Word Problems and Satisfiability Problems

We will use V to denote a countably infinite set of variables, and $T(\Omega, V)$ to denote the set of all *Ω-terms*, that is, terms over the signature Ω with variables in V. An equational theory E over the signature Ω is a set of (implicitly universally quantified) equations between Ω-terms. We use $s \equiv t$ to denote an equation between the terms s, t. For an equational theory E, the *word problem* is concerned with the validity in E of equations between Ω-terms. Equivalently, the word problem asks for the (un)satisfiability of the *disequation* $s \not\equiv t$ in E—where $s \not\equiv t$ is an abbreviation for the formula $\neg(s \equiv t)$. As usual, we write "$s =_E t$" to express that the formula $s \equiv t$ is valid in E. An equational theory E is *collapse-free* iff $x \neq_E t$ for all variables x and non-variable terms t.

[2] The qualification "potentially" is mandatory, of course, because we still need to impose some additional computability requirements on the theories to combine.

Given an Ω-term s, an Ω-algebra $\mathcal{A}$, and a valuation α (of the variables in s by elements of $\mathcal{A}$), we denote by $[\![s]\!]^{\mathcal{A}}_{\alpha}$ the interpretation of s in $\mathcal{A}$ under α. Also, if Σ is a subsignature of Ω, we denote by $\mathcal{A}^{\Sigma}$ the reduct of $\mathcal{A}$ to Σ. An Ω-algebra $\mathcal{A}$ is a *model* of E iff every equation in E is valid in $\mathcal{A}$. The equational theory E over the signature Ω defines an *Ω-variety*, i.e., the class of all models of E. When E is *non-trivial*, i.e., has models of cardinality greater than 1, its variety contains free algebras for any set of (free) generators. If $\mathcal{A}$ is a free algebra in E's Ω-variety with a set X of generators we say that $\mathcal{A}$ is *free in E over X*.

We are interested in *combined* equational theories, that is, equational theories E of the form $E := E_1 \cup E_2$, where E_1 and E_2 are equational theories over two signatures Σ_1 and Σ_2. We call the elements of $\Sigma := \Sigma_1 \cap \Sigma_2$, if any, *shared* symbols. We call 1-*symbols* the elements of Σ_1 and 2-*symbols* the elements of Σ_2. A term $t \in T(\Sigma_1 \cup \Sigma_2, V)$ is an *i-term* iff its *top symbol* $t(\epsilon) \in V \cup \Sigma_i$. Note that variables and terms t with $t(\epsilon) \in \Sigma_1 \cap \Sigma_2$ are both 1- and 2-terms. A subterm s of a 1-term t is an *alien subterm* of t iff it is not a 1-term and every proper superterm of s in t is a 1-term. Alien subterms of 2-terms are defined analogously. A term over the joint signature $\Sigma_1 \cup \Sigma_2$ is called a *shared* term if it is a Σ-term, and a *pure* term if it is a Σ_i-term for $i \in \{1, 2\}$, Similarly, an equation $s \equiv t$ is a *pure* equation if s and t are both Σ_i-terms for $i \in \{1, 2\}$.

A given disequation $s \not\equiv t$ between $(\Sigma_1 \cup \Sigma_2)$-terms s, t can be transformed into an equisatisfiable formula $\varphi_1 \wedge \varphi_2$, where φ_i is a conjunction of pure equations and disequations. This can be achieved by the usual *variable abstraction* process in which alien subterms are replaced by new variables (see the long version of [4] for a detailed description). Obviously, if $\varphi_1 \wedge \varphi_2$ is satisfiable in a model $\mathcal{A}$ of $E_1 \cup E_2$, each φ_i is then satisfiable in the reduct $\mathcal{A}^{\Sigma_i}$, which is a model of E_i $(i = 1, 2)$. However, if each φ_i is satisfiable in a model $\mathcal{A}_i$ of E_i, there may be no model of E in which $\varphi_1 \wedge \varphi_2$ is satisfiable. One case in which there always is one is described by the proposition below (see the long version of [4] for a proof).

Proposition 1. *Let $\mathcal{A}_i$ be a model of E_i, φ_i a first-order Σ_i-formula ($i = 1, 2$), and $\Sigma := \Sigma_1 \cap \Sigma_2$. Assume that ${\mathcal{A}_1}^{\Sigma}$ and ${\mathcal{A}_2}^{\Sigma}$ are both free in the same Σ-variety over respective sets of generators Y_1 and Y_2 with the same cardinality. If φ_i is satisfiable in $\mathcal{A}_i$ with the variables in $\mathcal{V}ar(\varphi_1) \cap \mathcal{V}ar(\varphi_2)$ taking distinct values over Y_i for $i = 1, 2$, then $\varphi_1 \wedge \varphi_2$ is satisfiable in a model of $E_1 \cup E_2$.*

As mentioned in the introduction, we will be interested in free models of E_1 and of E_2 whose reducts to their shared signature are themselves free. In general, the property of being a free algebra is not preserved under signature reduction. The problem is that the reduct of an algebra may need more generators than the algebra itself and these generators need not be free. Nonetheless, there are free algebras admitting reducts that are also free, although over a possibly larger set of generators. These algebras are models of equational theories that admit *constructors* in the sense defined in the next subsection.

2.1 Theories Admitting Constructors

In the following, Ω will be an at most countably infinite signature, and Σ a subset of Ω. We will fix a non-trivial equational theory E over Ω and define the *Σ-restriction of E* as $E^{\Sigma} := \{s \equiv t \mid s, t \in T(\Sigma, V) \text{ and } s =_E t\}$.

Definition 2 (Constructors). *The subsignature Σ of Ω is a* set of constructors for E *if for every Ω-algebra $\mathcal{A}$ free in E over a countably infinite set X, $\mathcal{A}^{\Sigma}$ is free in E^{Σ} over a set Y including X.*

As we will see, this new definition of constructors is a proper generalization of the definition given in [4], which also requires E^{Σ} to be collapse-free. Contrary to the one above, that definition does not require the generators of $\mathcal{A}^{\Sigma}$ to include those of $\mathcal{A}$; but this is always the case when E^{Σ} is collapse-free.

It is immediate that the whole signature Ω is a set of constructors for the theory E. Similarly, the empty signature is a set of constructors for E, as any model of E is free over its whole carrier in $E^{\emptyset}$, which is axiomatized by $\{v \equiv v \mid v \in V\}$. If E is the union of two theories over *disjoint* signatures, Σ_1, Σ_2 respectively, then each Σ_i is a set of constructors for E. This is not immediate, but it can be shown as a consequence of some results in [2].

In the following, we provide a more concrete, syntactic characterization of theories admitting constructors. For that we will introduce the concept of a *Σ-base*. But first, some more notation will be needed.

Given a subset G of $T(\Omega, V)$, we denote by $T(\Sigma, G)$ the set of terms over the "variables" G. More precisely, every member t of $T(\Sigma, G)$ is obtained from a term $s \in T(\Sigma, V)$ by replacing the variables of s with terms from G. We will denote such a term t by $s(\bar{r})$ where $\bar{r}$ is the tuple made, without repetitions, of the terms of G that replace the variables of s. Notice that this notation is consistent with the fact that $G \subseteq T(\Sigma, G)$. In fact, every $r \in G$ can be represented as $s(r)$ where s is a variable of V. Also notice that $T(\Sigma, V) \subseteq T(\Sigma, G)$ whenever $V \subseteq G$. In this case, every $s \in T(\Sigma, V)$ can be trivially represented as $s(\bar{v})$ where $\bar{v}$ are the variables of s.

Definition 3 (Σ-base). *A set $G \subseteq T(\Omega, V)$ is a* Σ-base of E *iff the following holds:*

1. *$V \subseteq G$.*
2. *For all $t \in T(\Omega, V)$, there is an $s(\bar{r}) \in T(\Sigma, G)$ such that $t =_E s(\bar{r})$.*
3. *For all $s_1(\bar{r}_1), s_2(\bar{r}_2) \in T(\Sigma, G)$,*

$$s_1(\bar{r}_1) =_E s_2(\bar{r}_2) \ \textit{ iff } \ s_1(\bar{v}_1) =_E s_2(\bar{v}_2),$$

where $\bar{v}_1, \bar{v}_2$ are fresh variables abstracting $\bar{r}_1, \bar{r}_2$ so that two terms in $\bar{r}_1, \bar{r}_2$ are abstracted by the same variable iff they are equivalent in E.

Theorem 4 (Characterization of constructors). *The signature Σ is a* set of constructors for E *iff E admits a Σ-base.*

The proof of the theorem—which can be found in [3]—provides a little more information than stated in the theorem.

Corollary 5. *Let G be a Σ-base of E, $\mathcal{A}$ an Ω-algebra free in E over a countably infinite set X, and α a bijective valuation of V onto X. Then $\mathcal{A}^\Sigma$ is free in E^Σ over the set $Y := \{[\![r]\!]^{\mathcal{A}}_\alpha \mid r \in G\}$, and $X \subseteq Y$.*

An interesting question is whether the condition that $X \subseteq Y$ in the definition of constructors is really needed. Does this condition always hold whenever the Σ-reduct of any algebra $\mathcal{A}$ free in E over the countably infinite set X is itself free? It can be easily shown that $\mathcal{A}^\Sigma$ can be free in E^Σ only over a set Y that is countably infinite. The question is: can Y always be chosen so that it includes X? When E^Σ is collapse-free, Y is unique and it does include X, so one needs not worry [4]. When E^Σ is not collapse-free, however, $\mathcal{A}^\Sigma$ may be free in E^Σ over more than one set of generators, not all of which include X.

For instance, consider the Ω-theory $E := \{g(g(x)) = x\}$ and let $\Sigma := \Omega$. Let $\mathcal{A}$ be an Ω-algebra free in E over some set X and α a bijective valuation of V onto X. It is easy to see that $\mathcal{A}$, which is free in E^Σ over X of course, is also free over the set $\{[\![g(v)]\!]^{\mathcal{A}}_\alpha \mid v \in V\}$, disjoint from X. Now, this example causes no problems because one can always choose $Y := X$ in this case. For the general case, however, the question remains open.

As shown in Cor. 5, a Σ-base of a theory E really denotes the set of free generators of a certain free model of E^Σ. Clearly, there may be many Σ-bases for the same theory E. For instance, if G is a Σ-base of E, any set obtained from G by replacing one of its terms by an E-equivalent term is also a Σ-base of E. Also, for any bijective renaming π of V onto itself, the set $\{\pi(r) \mid r \in G\}$ is a Σ-base of E as well.

One may wonder then if it is possible for E to have Σ-bases that are not just syntactic variants of one another. We know that this is not possible when E^Σ is collapse-free. In that case in fact, all Σ-bases, if any, denote the unique set Y over which the Σ-reduct of the infinitely generated free model of E is free. As it turns out, then E has a Σ-base $G_E(\Sigma, V)$ which is closed under bijective renaming of variables and under equivalence in E, and as such includes all the Σ-bases of E. In [4] and [14], where the definition of constructors includes the requirement that E^Σ be collapse-free, this maximal Σ-base is defined as follows:

$$G_E(\Sigma, V) := \{r \in T(\Omega, V) \mid r \neq_E t \text{ for all } t \in T(\Omega, V) \text{ with } t(\epsilon) \in \Sigma\}.$$

Modulo equivalence in E, $G_E(\Sigma, V)$ is made of Ω-terms whose top symbol is not in Σ, from which it is immediate that $G_E(\Sigma, V)$ is closed under bijective renaming and under equivalence in E. To summarize, the following holds for $G_E(\Sigma, V)$ [4].

Proposition 6. *Whenever E^Σ is collapse-free,*

- *every Σ-base of E, if any, is included in $G_E(\Sigma, V)$;*
- *Σ is a set of constructors for E iff $G_E(\Sigma, V)$ is a Σ-base of E.*

Examples of theories admitting (collapse-free) constructors can be found in [4]. We provide below an example of an equational theory admitting non-collapse-free constructors, that is, constructors in the more general sense of Definition 2. But first, it is instructive to look at at least one counterexample.

Let $E := \{f(g(x)) \equiv f(f(g(x)))\}$ and $\Sigma := \{f\}$. Since E^Σ is clearly collapse-free, we know that every Σ-base of E, if any, is included in the set $G_E(\Sigma, V)$ defined earlier. It is easy to see that $G_E(\Sigma, V) = V \cup \{g(t) \mid t \in T(\Omega, V)\}$ and that conditions (1) and (2) of Definition 3 hold for $G_E(\Sigma, V)$. However, condition (3) does not since $f(g(x)) =_E f(f(g(x)))$, although $f(y) \neq_E f(f(y))$. In conclusion, Σ is not a set of constructors for E.

Example 7. Consider the signature $\Sigma := \{\mathsf{0}, \mathsf{s}, \mathsf{p}, -\}$ and the equational theory E of the integers with zero, successor, predecessor, and unary minus, axiomatized by the equations:

$$x \equiv \mathsf{s}(\mathsf{p}(x)), \quad x \equiv \mathsf{p}(\mathsf{s}(x))$$
$$-\mathsf{0} \equiv \mathsf{0}, \quad -\mathsf{s}(x) \equiv \mathsf{p}(-x), \quad -\mathsf{p}(x) \equiv \mathsf{s}(-x), \quad -(-x) \equiv x.$$

The signature $\Sigma := \{\mathsf{0}, \mathsf{s}, \mathsf{p}\}$ is a set of constructors for E. This is proven in [3] by showing that the set $G := V \cup \{-v \mid v \in V\}$ is a Σ-base of E.

Many more examples of theories with constructors can be found in the usual axiomatizations of abstract data types. In the next subsection, we point out another, perhaps less obvious, class of examples for which our combination approach could provide fresh insights and results.

Normal Forms According to Definition 3, if a set G is a Σ-base of E, every Ω-term t is equivalent in E to a term $s(\bar{r}) \in T(\Sigma, G)$. We call $s(\bar{r})$ *a G-normal form of t in E*.[3] We say that a term t is *in G-normal form* if it is already of the form $t = s(\bar{r}) \in T(\Sigma, G)$. Because $V \subseteq G$, it is immediate that Σ-terms are in G-normal form, as are terms in G.

We will make use of normal forms in our combination procedure. In particular, we will consider normal forms that are computable in the following sense.

Definition 8 (Computable Normal Forms). *For any Σ-base G of E we say that* G-normal forms are computable for Σ and E *if there is a computable function $NF_E^\Sigma\colon T(\Omega, V) \longrightarrow T(\Sigma, G)$ such that $NF_E^\Sigma(t)$ is a G-normal form of t, i.e., $NF_E^\Sigma(t) =_E t$.*

Note that, unless E^Σ is collapse-free, the terms of G may as well start with a Σ-symbol themselves. This means that, for any given term t in G-normal form, it may not be possible to effectively identify those terms $\bar{r}$ of G such that $t = s(\bar{r})$ for some Σ-term s. Now, in the combination procedure shown in Sect. 3 sometimes we will need to first compute the normal form of a term and then decompose that into its components s and $\bar{r}$. To be able to do this it will be enough to assume (in addition to the computability of normal forms) that G is a recursive set, thanks to the proposition below.

[3] Notice that in general a term may have more than one G-normal form.

Proposition 9. *Let G be a Σ-base of E and $t \in T(\Sigma, G)$. If G is recursive, there is an effective way of computing a term $s(\bar{v}) \in T(\Sigma, V)$ and a sequence $\bar{r}$ of terms in G such that $t = s(\bar{r})$.*

2.2 Constructors and Modal Logics

For all normal modal logics [9], equivalence of formulae is a congruence relation on formulae that is closed under substitution [9]. For example, consider the basic modal logic K. Here, the signature Σ_{K} contains the Boolean operators ($\wedge$, $\vee$, $\neg$), the Boolean constant $\top$ (for truth), and the unary (modal) operator $\Box$. Equivalence of formulae in K can be axiomatized [10] by the equational theory E_{K}, which consists of the equational axioms for Boolean algebras, and the two additional equational axioms

$$\Box(x \wedge y) \equiv \Box(x) \wedge \Box(y) \quad \text{and} \quad \Box(\top) \equiv \top.$$

It is easy to see that satisfiability (and validity) of modal formulae in K is decidable iff the word problem for E_{K} is decidable. For example, a formula φ is valid iff $\varphi =_{E_{\mathsf{K}}} \top$. Since satisfiability in K is indeed decidable[4] the word problem for E_{K} is also decidable.

The problem of combining modal logics has been thoroughly investigated (see, e.g., [6, 8]). In particular, there are very general results on how decidability of the component logics transfers to their combination (called *fusion* in the literature). We are interested in the question of whether these combination results can also be obtained within our framework for combining decision procedures for the word problem. This line of research appears to be promising for the following two reasons.

First, it follows from results in [9] (Chap. 4.2) that equivalence in the fusion of two modal logics is axiomatized by the union of the equational theories axiomatizing equivalence in the component logics. In this union, the shared symbols are the Boolean symbols, i.e., $\wedge$, $\vee$, $\neg$, and $\top$. Since the axioms for Boolean algebras contain collapse axioms (e.g., $x \wedge x \equiv x$), it is clear that we will really need the generalized version of constructors introduced in this paper.

Second, the requirement that the reduct of the free algebra to the shared symbols be free is always satisfied for modal logics closed under the Boolean operations $\wedge$, $\vee$, and $\neg$. For example, let Σ be the subsignature of Σ_{K} that consists of $\wedge$, $\vee$, $\neg$, and $\top$. It is easy to show that the Σ-reduct ${\mathcal{A}_{\mathsf{K}}}^{\Sigma}$ of the E_{K}-free algebra $\mathcal{A}_{\mathsf{K}}$ over countably infinitely many generators is a countably infinite *atomless* Boolean algebra. Since the free Boolean algebra over countably infinitely many generators is also a countably infinite *atomless* Boolean algebra, and since all countably infinite atomless Boolean algebras are known to be isomorphic [7], we can deduce that the reduct ${\mathcal{A}_{\mathsf{K}}}^{\Sigma}$ is in fact free. For our combination method to apply, however, this is not sufficient. We need additional conditions; e.g., that normal forms are computable. Unfortunately, it is not even clear how a Σ-base

[4] In fact, it is a well-known PSPACE-complete problem.

could look like in this case. This would depend on an appropriate characterization of the generators of ${\mathcal{A}_\mathsf{K}}^{\Sigma}$, which appears to be a non-trivial (and, to the best of our knowledge, not yet solved) problem.

For this reason, we restrict our attention in the example below to a certain sublogic of K. Such a sublogic, which is not Boolean closed, is particularly interesting because the current combination results in modal logic are restricted to Boolean closed modal logics.

Example 10. Let us consider just the conjunctive fragment of K. In equational terms, this amounts to restricting the signature Σ_K to the subsignature $\Sigma^0_\mathsf{K} := \{\wedge, \top, \Box\}$ and consider only terms (i.e., modal formulae) built over this signature.

In [1], it is shown[5] that equivalence of such formulae is axiomatized by the theory E^0_K, which consists of the axioms

$$x \wedge (y \wedge z) \equiv (x \wedge y) \wedge z, \quad x \wedge y \equiv y \wedge x, \quad x \wedge x \equiv x, \quad x \wedge \top \equiv \top$$
$$\Box(x \wedge y) \equiv \Box(x) \wedge \Box(y), \quad \Box(\top) \equiv \top.$$

We claim that $\Sigma^0 := \{\wedge, \top\}$ is a set of constructors in our sense. In fact, the set

$$G := \{\Box^n(v) \mid n \geq 0 \text{ and } v \in V\}$$

can be shown to be a Σ^0-base of E^0_K. This is an easy consequence of the notion of concept-based normal form introduced in [1] and the characterization of equivalence proved in the same paper. The concept-based normal form of a formula is obtained by exhaustively applying the rewrite rules $\Box(x \wedge y) \rightarrow \Box(x) \wedge \Box(y)$, $\Box(\top) \rightarrow \top$, $x \wedge \top \rightarrow x$, $\top \wedge x \rightarrow x$. It is easy to see that this normal form can be computed in polynomial time, and that any formula in normal form is either $\top$ or a conjunction of elements of G. Thus, the concept-based normal form is also a G-normal form. Since the set G is obviously recursive and closed under variable renaming, the additional prerequisites (see below) for our combination approach to apply to E^0_K are satisfied as well.

Interestingly, if we consider the conjunctive fragment of the modal logic S_4 in place of K, we obtain a quite different behavior: the reduct of the free algebra to Σ^0 is not free (see [4]). This is surprising as, in the Boolean closed case, S_4 behaves like K in the sense that the reduct of the corresponding free algebra is still free (for the same reasons as for K).

2.3 Combination of Theories Sharing Constructors

To conclude this section we go back to the problem of combining theories and consider two non-trivial equational theories E_1, E_2 with respective signatures Σ_1, Σ_2 such that $\Sigma := \Sigma_1 \cap \Sigma_2$ is a set of constructors for E_1 and for E_2. Moreover, we assume that ${E_1}^{\Sigma} = {E_2}^{\Sigma}$.

[5] Note, however, that [1] employs description logic syntax rather than modal logic syntax for formulae.

For $i = 1, 2$, let $\mathcal{A}_i$ be a Σ_i-algebra free in E_i over some countably infinite set X_i, and $Y_i := \{[\![r]\!]^{\mathcal{A}_i}_{\alpha_i} \mid r \in G_i\}$ where G_i is any Σ-base of E_i and α_i any bijection of V onto X_i. From Prop. 1 and Cor. 5, we then have the following:

Proposition 11. *Let φ_1, φ_2 be two first-order formulae of respective signature Σ_1, Σ_2. If φ_i is satisfiable in $\mathcal{A}_i$ with the elements of $\mathcal{V}ar(\varphi_1) \cap \mathcal{V}ar(\varphi_2)$ taking distinct values over Y_i for $i = 1, 2$, then $\varphi_1 \wedge \varphi_2$ is satisfiable in $E_1 \cup E_2$.*

An immediate consequence of this result is that the theory $E_1 \cup E_2$ above is non-trivial. To see that, since φ_1 and φ_2 in the proposition are arbitrary formulae, it is enough to take both of them to be the disequation $x \not\equiv y$ between two distinct variables.

In the rest of the paper we show that, under the above assumptions on E_1 and E_2, the combined theory $E_1 \cup E_2$ in fact has much stronger properties: whenever normal forms are computable for Σ and E_i $(i = 1, 2)$ with respect to a recursive Σ-base closed under renaming, the decidability of the word problem is a modular property, as is the property of being a set of constructors.

3 A Combination Procedure for the Word Problem

In the following, we present a decision procedure for the word problem in an equational theory of the form $E_1 \cup E_2$ where each E_i is an equational theory with decidable word problem. Such a procedure will be obtained as modular combination of the procedures deciding the word problem for E_1 and for E_2. We will restrict our attention to equational theories E_1, E_2 that satisfy the following conditions for $i = 1, 2$:

1. *E_i is a non-trivial equational theory over the (countable) signature Σ_i;*
2. *$\Sigma := \Sigma_1 \cap \Sigma_2$ is a set of constructors for E_i and ${E_1}^{\Sigma} = {E_2}^{\Sigma}$;*
3. *E_i admits a Σ-base G_i closed under bijective renaming of V;*
4. *G_i is recursive and G_i-normal forms are computable for Σ and E_i;*
5. *the word problem for E_i is decidable.*

As already mentioned, the word problem for $E := E_1 \cup E_2$ can be reduced to the satisfiability problem (in E) for disequations of the form $s_0 \not\equiv t_0$, where s_0, t_0 are $(\Sigma_1 \cup \Sigma_2)$-terms. By variable abstraction it is possible to transform any such disequation into a a set $AS(s_0 \not\equiv t_0)$ consisting of pure equations and a disequation between two variables such that $s_0 \not\equiv t_0$ and $AS(s_0 \not\equiv t_0)$ are "equivalent" in a sense to be made more precise below. The set $AS(s_0 \not\equiv t_0)$ is what we call an *abstraction system*. To define abstraction systems formally we will need some more notation.

To start with, we will use finite sets of formulae in place of conjunctions of such formulae, and say that a set S of formulae is satisfiable in a theory iff the conjunction of its elements is satisfiable in that theory. Now, let T be a set of equations of the form $v \equiv t$ where $v \in V$ and $t \in T(\Sigma_1 \cup \Sigma_2, V) \setminus V$. The relation $\prec$ on T is defined as follows: $(u \equiv s) \prec (v \equiv t)$ iff $v \in \mathcal{V}ar(s)$.

By $\prec^+$ we denote the *transitive* and by $\prec^*$ the *reflexive-transitive closure* of $\prec$. The relation $\prec$ is *acyclic* if there is no equation $v \equiv t$ in T such that $(v \equiv t) \prec^+ (v \equiv t)$.

Definition 12 (Abstraction System). *A set $\{x \not\equiv y\} \cup T$ is an* abstraction system with disequation $x \not\equiv y$ *iff $x, y \in V$ and the following holds:*

1. *T is a finite set of equations of the form $v \equiv t$ where $v \in V$ and $t \in (T(\Sigma_1, V) \cup T(\Sigma_2, V)) \setminus V$;*
2. *the relation $\prec$ on T is acyclic;*
3. *for all $(u \equiv s), (v \equiv t) \in T$,*
   (a) if $u = v$ then $s = t$;
   (b) if $(u \equiv s) \prec (v \equiv t)$ and $s \in T(\Sigma_i, V)$ with $i \in \{1, 2\}$ then $t \notin T(\Sigma_i, V)$.

The above is a generalization of the definition of abstraction system in [4] in that now the right-hand side of any equation in T can start with a shared symbol. As before, Condition (2) implies that for all $(u \equiv s), (v \equiv t) \in T$, if $(u \equiv s) \prec^* (v \equiv t)$ then $u \notin \mathcal{V}ar(t)$; Condition (3a) implies that a variable cannot occur as the left-hand side of more than one equation of T; Condition (3b) implies, together with Condition (1), that the elements of every $\prec$-chain of T have *strictly* alternating signatures $(\ldots, \Sigma_1, \Sigma_2, \Sigma_1, \Sigma_2, \ldots)$. In particular, when Σ_1 and Σ_2 have a non-empty intersection Σ, Condition (3b) entails that if $(u \equiv s) \prec (v \equiv t)$ neither s nor t can be a Σ-term: one of the two must contain symbols from $\Sigma_1 \setminus \Sigma$ and the other must contain symbols from $\Sigma_2 \setminus \Sigma$.

Proposition 13. *The set $S := AS(s_0 \not\equiv t_0)$ is an abstraction system. Furthermore, where $\bar{v}$ is the tuple that collects all the variables in the left-hand side of an equations of S, the formula $\exists \bar{v}.S \leftrightarrow (s_0 \not\equiv t_0)$ is logically valid.*

Every abstraction system $\{x \not\equiv y\} \cup T$ induces a finite graph $\mathcal{G}_S := (T, \prec)$ whose set of *edges* consists of all pairs $(n_1, n_2) \in T \times T$ such that $n_1 \prec n_2$. According to Definition 12, $\mathcal{G}_S$ is in fact a directed acyclic graph (or *dag*). Assuming the standard definition of path between two nodes and of length of a path in a dag, the *height* $\mathsf{h}(n)$ of the node n is the maximum of the lengths of all the paths in the dag that end with n.[6]

In the previous section, we would have represented the normal form of a term in $T(\Sigma_i, V)$ $(i = 1, 2)$ as $s(\bar{q})$ where s was a term in $T(\Sigma, V)$ and $\bar{q}$ a tuple of terms in G_i. Considering that G_i contains V, we will now use a more descriptive notation. We will distinguish the variables in $\bar{q}$ from the non-variables terms and write $s(\bar{y}, \bar{r})$ instead, where $\bar{y}$ collects the elements of $\bar{q}$ that are in V and $\bar{r}$ those that are in $G_i \setminus V$.

The combination procedure described in Fig. 1 decides the word problem for the theory $E := E_1 \cup E_2$ by deciding the satisfiability in E of disequations of the form $s_0 \not\equiv t_0$ where s_0, t_0 are $(\Sigma_1 \cup \Sigma_2)$-terms. During the execution of the procedure, the set S of formulae on which the procedure works is repeatedly modified by the application of one of the derivation rules defined in Fig. 2. We

[6] Since $\mathcal{G}_S$ is acyclic and finite, this maximum exists.

Input: $(s_0, t_0) \in T(\Sigma_1 \cup \Sigma_2, V) \times T(\Sigma_1 \cup \Sigma_2, V)$.

1. Let $S := AS(s_0 \not\equiv t_0)$.
2. Repeatedly apply (in any order) **Coll1**, **Coll2**, **Ident**, **Simpl**, **Shar1**, **Shar2** to S until none of them is applicable.
3. Succeed if S has the form $\{v \not\equiv v\} \cup T$ and fail otherwise.

Fig. 1. The Combination Procedure.

Coll1 $$\frac{T \quad u \not\equiv v \quad x \equiv t[y] \quad y \equiv r}{T[x/r] \quad (u \not\equiv v)[x/y] \quad y \equiv r}$$

if $t \in T(\Sigma_i, V)$ and $y =_{E_i} t$ for $i = 1$ or $i = 2$.

Coll2 $$\frac{T \quad x \equiv t[y]}{T[x/y]}$$

if $t \in T(\Sigma_i, V)$ and $y =_{E_i} t$ for $i = 1$ or $i = 2$, and there is no $(y \equiv r) \in T$.

Ident $$\frac{T \quad x \equiv s \quad y \equiv t}{T[x/y] \quad y \equiv t}$$

if $s, t \in T(\Sigma_i, V)$ and $s =_{E_i} t$ for $i = 1$ or $i = 2$, $x \neq y$, and $\mathsf{h}(x \equiv s) \leq \mathsf{h}(y \equiv t)$.

Simpl $$\frac{T \quad x \equiv t}{T} \quad \text{if } x \notin \mathcal{V}ar(T).$$

Shar1 $$\frac{T \quad u \not\equiv v \quad x \equiv t \quad \bar{y}_1 \equiv \bar{r}_1}{T[x/s(\bar{y}, \bar{z})[\bar{y}_1/\bar{r}_1]] \quad \bar{z} \equiv \bar{r} \quad u \not\equiv v \quad x \equiv s(\bar{y}, \bar{r}) \quad \bar{y}_1 \equiv \bar{r}_1}$$

if (a) $x \in \mathcal{V}ar(T)$,
(b) $t \in T(\Sigma_i, V) \setminus G_i$ for $i = 1$ or $i = 2$,
(c) $NF^{\Sigma}_{E_i}(t) = s(\bar{y}, \bar{r}) \in T(\Sigma, G_i) \setminus V$,
(d) $\bar{r}$ nonempty and $\bar{r} \subseteq G_i \setminus T(\Sigma, V)$,
(e) $\bar{z}$ fresh variables with no repetitions,
(f) $\bar{y}_1 \subseteq \mathcal{V}ar(s(\bar{y}, \bar{r}))$ and $(x \equiv s(\bar{y}, \bar{r})) \prec (y \equiv r)$ for no $(y \equiv r) \in T$.

Shar2 $$\frac{T \quad u \not\equiv v \quad x \equiv t \quad \bar{y}_1 \equiv \bar{r}_1}{T[x/s[\bar{y}_1/\bar{r}_1]] \quad u \not\equiv v \quad x \equiv s[\bar{y}_1/\bar{r}_1] \quad \bar{y}_1 \equiv \bar{r}_1}$$

if (a) $x \in \mathcal{V}ar(T)$,
(b) $t \in T(\Sigma_i, V) \setminus G_i$ for $i = 1$ or $i = 2$,
(c) $NF^{\Sigma}_{E_i}(t) = s \in T(\Sigma, V) \setminus V$,
(d) $\bar{y}_1 \subseteq \mathcal{V}ar(s)$,
(e) $\bar{r}_1 \subseteq G_\iota$ with $\iota \in \{1, 2\} \setminus \{i\}$, and $(x \equiv s) \prec (y \equiv r)$ for no $(y \equiv r) \in T$.

Fig. 2. The Derivation Rules.

describe these rules in the style of a sequent calculus. The premise of each rule lists all the formulae in S before the application of the rule, where T stands for all the formulae not explicitly listed. The conclusion of the rule lists all the formulae in S after the application of the rule.

The procedure and the rules are almost identical to those given in [4]. The only difference is that the rules **Shar1** and **Shar2** have a different set of preconditions to account for the generalization of the notion of normal form caused by the new definition of constructors.

As before, **Coll1** and **Coll2** remove from S collapse equations that are valid in E_1 or E_2, while **Ident** identifies any two variables equated to equivalent Σ_i-terms and then discards one of the corresponding equations. The ordering restriction in the precondition of **Ident** is on the heights that the two equations involved have in the dag induced by S. It is there to prevent the creation of cycles in the relation $\prec$ over S. We have used the notation $t[y]$ to express that the variable y occurs in the term t, and the notation $T[x/t]$ to denote the set of formulae obtained by substituting every occurrence of the variable x by the term t in the set T.

The rule **Simpl** reduces clutter in S by eliminating those equations that have become unreachable along a $\prec$-path from the initial disequation because of the application of previous rules.

The rules **Shar1** and **Shar2**, which only apply when Σ_1 and Σ_2 are non-disjoint, are used in essence to propagate the constraint information represented by shared terms. To do that, the rules replace the right-hand side t of an equation $x \equiv t$ by its normal form, and then plug the "shared part" of the normal form in all those equations whose right-hand side contains x. In the description of the rules, an expression like $\bar{z} \equiv \bar{r}$ denotes the set $\{z_1 \equiv r_1, \ldots, z_n \equiv r_n\}$ where $\bar{z} = (z_1, \ldots, z_n)$ and $\bar{r} = (r_1, \ldots, r_n)$, and $s(\bar{y}, \bar{z})$ denotes the term obtained from $s(\bar{y}, \bar{r})$ by replacing the subterm r_j with z_j for each $j \in \{1, \ldots, n\}$. This notation also accounts for the possibility that t reduces to a non-variable term of G_i. In that case, s will be a variable, $\bar{y}$ will be empty, and $\bar{r}$ will be a tuple of length 1. Substitution expressions containing tuples are to be interpreted accordingly; also, tuples are sometimes used to denote the set of their components.

We make one assumption on **Shar1** and **Shar2** which is not explicitly listed in their preconditions. We assume that $NF^{\Sigma}_{E_i}$ $(i = 1, 2)$ is such that, whenever the set $V_0 := \mathcal{V}ar(NF^{\Sigma}_{E_i}(t)) \setminus \mathcal{V}ar(t)$ is not empty,[7] each variable in V_0 is fresh with respect to the current set S. As explained in [3], this assumption can be made without loss of generality whenever G_i is closed under renaming of variables.

By requiring that $\bar{r}$ be non-empty, **Shar1** excludes the possibility that the normal form of the term t is a shared term. It is **Shar2** that deals with this case. The reason for a separate case is that we want to preserve the property that every $\prec$-chain is made of equations with alternating signatures (cf. Definition 12(3b)). When the equation $x \equiv t$ has immediate $\prec$-successors, the replacement of t by the Σ-term s may destroy the alternating signatures property because $x \equiv s$,

[7] This might happen if E_i is non-regular because then Definition 8 does not necessarily entail that all the variables of $NF^{\Sigma}_{E_i}(t)$ occur in t.

which is both a Σ_1- and a Σ_2-equation, may inherit some of these successors from $x \equiv t$.[8] **Shar2** restores this property by merging into $x \equiv s$ all of its immediate successors, if any. Condition (e) in **Shar2** makes sure that the tuple $\bar{y}_1 \equiv \bar{r}_1$ collects all these successors. The replacement of $\bar{y}_1$ by $\bar{r}_1$ in **Shar1** is done for similar reasons. In **Shar2**, the restriction that the terms in $\bar{r}_1$ be elements of G_i is necessary to ensure termination, as is the condition $x \in \mathcal{V}ar(T)$ in both rules.

A Sketch of the Correctness Proof

The total correctness of the combination procedure can be proven more or less in the same way as in [4]. We can first show that an application of one of the rules of Fig. 2 transforms abstraction systems into abstraction systems, preserves satisfiability, and leads to a decrease with respect to a certain well-founded ordering. This ordering can be obtained as follows: every node in the dag induced by the abstraction system S is associated with a pair (h, r), where h is the height of the node, r is 1 if the right-hand side of the node is neither in G_1 nor in G_2, and 0 otherwise. The abstraction system S is associated with the multiset $M(S)$ consisting of all these pairs. Let $\sqsupset$ be the multiset ordering induced by the lexicographic ordering on pairs.

Lemma 14. *Assume that S' is obtained from the abstraction system S by an application of one of the rules of Fig. 2. Then the following holds:*

1. $M(S) \sqsupset M(S')$.
2. *S' is an abstraction system.*
3. *$\exists \bar{v}.S \leftrightarrow \exists \bar{v}'.S'$ is valid in E, where $\bar{v}$ lists all the left-hand side variables of S and $\bar{v}'$ the left-hand side variables of S'.*

Since the multiset ordering $\sqsupset$ is well-founded, the first point of the lemma implies that the derivation rules can be applied only finitely many times. Given that the preconditions of each rule are effective because of the computability assumptions on the component theories and of Prop. 9, we can then conclude that the combination procedure halts on all inputs. The last point of the lemma together with Prop. 13 implies that the procedure is sound, that is, if it succeeds on an input (s_0, t_0), then $s_0 =_E t_0$. The second point implies that the final system obtained after the termination of the procedure is an abstraction system, which plays an important rôle in the completeness proof. We can prove that the combination procedure is complete, that is, succeeds on an input (s_0, t_0) whenever $s_0 =_E t_0$, by showing that Prop. 11 can be applied (see [3] for details). To sum up, this shows the overall correctness of the procedure, and thus:

Lemma 15. *Under the given assumptions, the word problem for $E := E_1 \cup E_2$ is decidable.*

[8] As explained above, we assume that the variables in $\mathcal{V}ar(s) \setminus \mathcal{V}ar(t)$ do not occur in the abstraction system. Thus, the equations in $\bar{y}_1 \equiv \bar{r}_1$ are in fact successors of $x \equiv t$.

The corresponding result in [4] is indeed a corollary of this one. The difference there is that we have the additional restriction that ${E_i}^{\Sigma}$ is collapse-free and we use the largest Σ-base of E_i, namely $G_{E_i}(\Sigma, V)$, instead of an arbitrary one ($i = 1, 2$). In [4], we do not *explicitly* assume that $G_{E_i}(\Sigma, V)$ is closed under renaming. But this is always the case, as we mentioned earlier. Also, we do not postulate that $G_{E_i}(\Sigma, V)$ is recursive because that is always the case whenever $G_{E_i}(\Sigma, V)$-normal forms are computable for Σ and E_i.

Similarly to [4], the decidability result of Lemma 15 is actually extensible to the union of any (finite) number of theories, all (pairwise) sharing the same signature Σ and satisfying the same properties as E_1 and E_2 above. The reason is that, again, all needed properties are modular with respect to theory union.

Theorem 16. *For all theories E_1, E_2 satisfying assumptions(1)-(5) stated at the beginning of Sect. 3, the following holds:*

1. *Σ is a set of constructors for $E := E_1 \cup E_2$ and $E^{\Sigma} = {E_1}^{\Sigma} = {E_2}^{\Sigma}$.*
2. *E admits a Σ-base G^* closed under bijective renaming of V;*
3. *G^* is recursive and G^*-normal forms are computable for Σ and E;*
4. *the word problem for E is decidable.*

The proof of the first three points is still quite involved (see [3] for details) but somewhat simpler than the corresponding proof in [4]. It depends on the explicit construction of the set G^*, given below. There, for all terms $r \in T(\Sigma_1 \cup \Sigma_2, V)$, we denote by $\hat{r}$ the pure term obtained from r by abstracting its alien subterms.

Definition 17. *The set G^* is inductively defined as follows:*

1. *Every variable is an element of G^*, that is, $V \subseteq G^*$.*
2. *Assume that $r(\bar{v}) \in G_i \setminus V$ for $i \in \{1, 2\}$ and $\bar{r}$ is a tuple of elements of G^* with the same length as $\bar{v}$ such that the following holds:*
   (a) $r(\bar{v}) \neq_E v$ for all variables $v \in V$;
   (b) $\hat{r}_k \notin T(\Sigma_i, V)$ for all non-variable components r_k of $\bar{r}$;
   (c) $r_k \neq_E r_\ell$ if r_k, r_ℓ occur at different positions in the tuple $\bar{r}$.
   Then $r(\bar{r}) \in G^*$.

Notice that every non-collapsing term of G_i is in G^* for $i = 1, 2$ because the components of $\bar{r}$ in the definition above can also be variables. Every non-variable element r of G^* then has a stratified structure. Each layer of r is made of terms all belonging to G_1 or to G_2. Moreover, if the terms in one layer are in G_i, then the terms in the layer above and below, if any, are not in G_i.

4 Future Work

The results presented in this paper are preliminary in two respects. First, they depend on two *technical restrictions* for which we do not yet know whether they are necessary. One is the requirement in the definition of constructors that $X \subseteq Y$ (used to prove the completeness of the combination procedure); the other is the requirement that the Σ-bases employed in the combination procedure be closed under renaming (used in the derivation rules and to prove the modularity results). We are trying to find out whether these restrictions can be removed, or there is a deeper, non-technical, reason for them.

Second, in order to demonstrate the power of our new combination procedure, we intend to investigate more thoroughly its applicability to the combination of decision procedures for modal logics. This probably depends on a deep understanding of the structure of the free algebras corresponding to the modal logics in question.

References

1. F. Baader and P. Narendran. Unification of concept terms in description logics. In H. Prade, editor, *Proc. of ECAI-98*, pages 331–335. John Wiley & Sons Ltd, 1998.
2. F. Baader and K. U. Schulz. Combination of constraint solvers for free and quasi-free structures. *Theoretical Computer Science*, 192:107–161, 1998.
3. F. Baader and C. Tinelli. Combining equational theories sharing non-collapse-free constructors. TR 99-13, Dept. of Computer Science, University of Iowa, 1999.
4. F. Baader and C. Tinelli. Deciding the word problem in the union of equational theories sharing constructors. In P. Narendran and M. Rusinowitch, editors, *Proc. of RTA'99*, volume 1631 of *LNCS*, pages 175–189. Springer-Verlag, 1999. Longer version available as TR UIUCDCS-R-98-2073, Dept. of Computer Science, University of Illinois at Urbana-Champaign, 1998.
5. E. Domenjoud, F. Klay, and Ch. Ringeissen. Combination techniques for non-disjoint equational theories. In A. Bundy, editor, *Proc. of CADE-12*, volume 814 of *LNAI*, pages 267–281. Springer-Verlag, 1994.
6. E. Hemaspaandra. Complexity transfer for modal logic (extended abstract). In *Proc, of LICS'94*, pages 164–173, 1994. IEEE Computer Society Press.
7. S. Koppelberg. General theory of Boolean algebras. In J. Monk, editor, *Handbook of Boolean Algebras*. North-Holland, Amsterdam, The Netherlands, 1988.
8. M. Kracht and F. Wolter. Simulation and transfer results in modal logic: A survey. *Studia Logica*, 59:149–177, 1997.
9. M. Kracht. *Tools and Techniques in Modal Logic*. Elsevier, Amsterdam, The Netherlands, 1999.
10. E. J. Lemmon. Algebraic semantics for modal logics. *J. of Symbolic Logic*, 31(1):46–65, 1966.
11. T. Nipkow. Combining matching algorithms: The regular case. In N. Dershowitz, editor, *Proc. of RTA'89*, volume 335 of *LNCS*, pages 343–358. Springer-Verlag, 1989.
12. Don Pigozzi. The join of equational theories. *Colloquium Mathematicum*, 30(1):15–25, 1974.
13. Manfred Schmidt-Schauß. Combination of unification algorithms. *J. of Symbolic Computation*, 8(1–2):51–100, 1989.
14. C. Tinelli and Ch. Ringeissen. Non-disjoint unions of theories and combinations of satisfiability procedures: First results. TR UIUCDCS-R-98-2044, Dept. of Computer Science, University of Illinois at Urbana-Champaign, 1998.