Combining Equational Theories Sharing Non-Collapse-Free Constructors

Franz Baader[1]
baader@informatik.rwth-aachen.de

Cesare Tinelli
tinelli@cs.uiowa.edu

Department of Computer Science
University of Iowa
Report No. 99-13

October 1999

[1]Address: Lehr- und Forschungsgebiet Theoretische Informatik, RWTH Aachen, Ahornstraße 55, 52074 Aachen, Germany.

Combining Equational Theories Sharing Non-Collapse-Free Constructors

Franz Baader
Lehr- und Forschungsgebiet Theoretische Informatik
RWTH Aachen
Ahornstraße 55, 52074 Aachen, Germany
baader@informatik.rwth-aachen.de

Cesare Tinelli
Department of Computer Science
University of Iowa
14 McLean Hall, Iowa City, IA 52242 – USA
tinelli@cs.uiowa.edu

October 1999

Abstract

In a previous work, we describe a method to combine decision procedures for the word problem for theories sharing constructors. One of the requirements of our combination method is that constructors be *collapse-free*. This paper removes that requirement by modifying the method so that it applies to non-collapse-free constructors as well. This broadens the scope of our combination results considerably, for example in the direction of equational theories corresponding to modal logics.

Keywords: word problem, combination of decision procedures, constructors.

Contents

1 Introduction **3**

2 Formal Preliminaries **4**

3 Combining Non-Disjoint Equational Theories **6**

3.1 Theories Admitting Constructors 7

3.2 Constructors and Modal Logics 15

3.3 Combination of Theories Sharing Constructors 17

4 A Combination Procedure for the Word Problem **18**

4.1 Abstraction Systems 18

4.2 The Combination Procedure 22

4.3 The Correctness Proof 26

5 Modularity of Constructors **38**

6 Conclusion and Open Questions **50**

1 Introduction

In [BT99] 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 [Pig74]. 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 [BT99], which was obtained as a refinement of one first introduced in [TR98] and generalized that in [DKR94], was build around the observation that some equational theories E 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 of constructors in [BT99], 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 report, 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 [BT99] continue to hold without it.

In [BT99] 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 report, 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 to expand considerably 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 a conservative extension of a same Σ-theory can be combined with out 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 Section 3.2. 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

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

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

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, prove its correctness, and show how that leads to exactly the same results given in [BT99], but of course with the wider scope provided by the new definition of constructors.

2 Formal Preliminaries

A (functional) *signature* is a set of function symbols with an associated arity. Throughout the report we will consider only countable signatures. We will denote by V a fixed countably infinite set of variables and for any signature Σ, we will denote by $T(\Sigma, V)$ the set of Σ-terms over V. We will use the symbols r, s, t to denote terms, and the symbols x, y, u, v, w, z to denote variables. With a common abuse of notation we will also use x, y, u, v, w, z as the actual variables in our examples.

If t is a term, $\mathcal{V}ar(t)$ will denote the set of all variables occurring in t. Similarly, if φ is a formula, $\mathcal{V}ar(\varphi)$ will denote the set of free variables of φ.

Where $\bar{v}$ is a tuple of variables without repetition, we will write $t(\bar{v})$ to mean that $\bar{v}$ lists *all* the variables of t. When convenient, we will treat a tuple $\bar{t}$ of terms as the set of its elements.

For any functional signature Σ, a *quantifier-free formula* is a Boolean combination of Σ-equations, i.e., of formulae of the form $s \equiv t$, where $\equiv$ denotes the equality predicate and s, t are terms in $T(\Sigma, V)$. We use the abbreviation $s \not\equiv t$ to denote the *disequation* $\neg(s \equiv t)$.

An equational theory E with signature Σ is a set of universally quantified equations between Σ-terms. As customary, we will omit the universal quantifiers; for example, we will denote the equational theory C axiomatizing the commutativity of the binary function symbol f by $C := \{f(x, y) \equiv f(y, x)\}$ instead of $C := \{\forall x, y. f(x, y) \equiv f(y, x)\}$.

As usual, we say that a formula φ is *valid* in E and write $E \models \varphi$ iff φ holds in all models of E, i.e., iff for all Σ-algebras $\mathcal{A}$ that satisfy E (are a *model* of E) and all valuations α of the free variables of φ by elements of $\mathcal{A}$ we have $\mathcal{A}, \alpha \models \varphi$. Since a formula is valid in E iff its negation is unsatisfiable in E, we can turn the validity problem for E into an equivalent *satisfiability problem:* we know that a formula φ is not valid in E iff there exist a Σ-model $\mathcal{A}$ of E and a valuation α such that $\mathcal{A}, \alpha \models \neg\varphi$.

Given a function symbol $f \in \Sigma$ and a Σ-algebra $\mathcal{A}$, we denote by $f^{\mathcal{A}}$ the interpretation of f in $\mathcal{A}$. This notation can be extended to terms in the obvious way: if s is a Σ-term containing n distinct variables, then we denote by $s^{\mathcal{A}}$ the n-ary term function induced by the term s in $\mathcal{A}$. Given a Σ-term s, a Σ-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 the term s in $\mathcal{A}$ under the valuation α. Using the term function induced by s, this interpretation of s can also be written as $[\![s]\!]^{\mathcal{A}}_{\alpha} = s^{\mathcal{A}}(\bar{a})$, where $\bar{a}$ is the tuple of values which α assigns to the variables in s.

For an equational theory E, the *word problem* is concerned with the validity in E of quantifier-free formulae of the form $s \equiv t$. Equivalently, the word problem asks for the (un)satisfiability of the *disequation* $s \not\equiv t$ in E. As usual, we often write "$s =_E t$" to express that the formula $s \equiv t$ is valid in E.

An equational theory E is *non-trivial* if it admits models of cardinality greater than 1; it is *collapse-free* iff $x \neq_E t$ for all variables x and non-variable terms t. It is easy to see that when E is non-trivial, $x =_E t$ only if x is a variable of t.

The equational theory E over the signature Σ defines a Σ-*variety*, i.e., the class of all the models of E. When E is non-trivial this variety contains free algebras for any set of generators. We will call these algebras *E-free algebras.* More precisely, if $\mathcal{A}$ is a free algebra in E's Σ-variety with a set X of free generators we will say that $\mathcal{A}$ is *free in E over X*, or also, that $\mathcal{A}$ is a *free model of E over X*.

The following is a well-known characterization of free algebras (see, e.g., [Hod93]):

Proposition 2.1 *Let E be an equational theory over Σ, $\mathcal{A}$ a Σ-algebra, and X a subset of $\mathcal{A}$'s carrier. Then, $\mathcal{A}$ is free in E over X iff the following holds:*

1. *$\mathcal{A}$ is a model of E;*
2. *X generates $\mathcal{A}$;*
3. *for all $s, t \in T(\Sigma, V)$, if $\mathcal{A}, \alpha \models s \equiv t$ for some injection α of $\mathcal{V}ar(s \equiv t)$ into X, then $s =_E t$.*

We will use another well known properties of free algebras.

Proposition 2.2 *Let $\mathcal{B}_1, \mathcal{B}_2$ be two algebras free in the same Σ-variety over respective sets Y_1, Y_2 of the same cardinality. Then, any bijection of Y_1 onto Y_2 extends to an isomorphism of $\mathcal{B}_1$ onto $\mathcal{B}_2$.*

In this report, 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 (not necessarily disjoint) functional signatures Σ_1 and Σ_2. The elements of $\Sigma_1 \cap \Sigma_2$ are called *shared* symbols. For any term t let $t(\epsilon)$ denote the top symbol of t. A term $t \in T(\Sigma_1 \cup \Sigma_2, V)$ is an *i-term* iff $t(\epsilon) \in V \cup \Sigma_i$, i.e., if it is a variable or has the form $t = f(t_1, ..., t_n)$ for some Σ_i-symbol f $(i = 1, 2)$. Notice 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. For $i = 1, 2$, an i-term s is *pure* iff it is a Σ_i-term. An equation $s \equiv t$ is pure iff there is an i such that both s and t are pure i-terms.

For some proofs, it will be important to know the number of "signature changes" that occur in a $(\Sigma_1 \cup \Sigma_2)$-term. This is formalized by the concept of rank.

Definition 2.3 (Rank) *Let t be a term in $T(\Sigma_1 \cup \Sigma_2, V)$ and $Aln(t)$ the set of all alien subterms of t. The* rank *of t (w.r.t. Σ_1 and Σ_2) is defined as follows.*

$$rank(t) := \begin{cases} 0 & \text{if } t \text{ is pure,} \\ 1 + \max\{rank(r) \mid r \in Aln(t)\} & \text{otherwise.} \end{cases}$$

3 Combining Non-Disjoint Equational Theories

Since the union of equational theories with decidable word problem need not have a decidable word problem, one needs appropriate restrictions on the theories to be combined. In this section we introduce such restrictions, and establish some useful properties of theories satisfying them. In particular, Proposition 3.4 will play a crucial rôle in the proof of completeness of the combination procedure.

In the following, given an Ω-algebra $\mathcal{A}$ and a subset Σ of Ω, we will denote by $\mathcal{A}^\Sigma$ the reduct of $\mathcal{A}$ to the subsignature Σ.

Definition 3.1 (Fusion) *A $(\Sigma_1 \cup \Sigma_2)$-algebra $\mathcal{F}$ is a* fusion *of a Σ_1-algebra $\mathcal{A}_1$ and a Σ_2-algebra $\mathcal{A}_2$ iff $\mathcal{F}^{\Sigma_1}$ is Σ_1-isomorphic to $\mathcal{A}_1$ and $\mathcal{F}^{\Sigma_2}$ is Σ_2-isomorphic to $\mathcal{A}_2$.*

In essence, a fusion of $\mathcal{A}_1$ and $\mathcal{A}_2$, if it exists, is an algebra that is identical to $\mathcal{A}_1$ when seen as a Σ_1-algebra, and identical to $\mathcal{A}_2$ when seen as a Σ_2-algebra. Clearly, two algebras $\mathcal{A}_1$ and $\mathcal{A}_2$ do not necessarily admit a fusion. In [BT99], it is shown that they do exactly when their reducts to their common signature $\Sigma := \Sigma_1 \cap \Sigma_2$ are isomorphic. More precisely, [BT99] shows that every (Σ-)isomorphism of ${\mathcal{A}_1}^\Sigma$ onto ${\mathcal{A}_2}^\Sigma$ induces a *canonical* fusion of $\mathcal{A}_1$ and $\mathcal{A}_2$.

Proposition 3.2 *Let $\mathcal{A}$ be a Σ_1-algebra, $\mathcal{B}$ a Σ_2-algebra, and $\Sigma := \Sigma_1 \cap \Sigma_2$. For every isomorphism h of ${\mathcal{A}_1}^\Sigma$ onto ${\mathcal{A}_2}^\Sigma$, there is a fusion $\mathcal{F}_h$ of $\mathcal{A}_1$ and $\mathcal{A}_2$ such that*

- *the identity function of $\mathcal{A}_2$ is a Σ_2-isomorphism of $\mathcal{A}_2$ onto ${\mathcal{F}_h}^{\Sigma_2}$,*
- *h is a Σ_1-isomorphism of $\mathcal{A}_1$ onto ${\mathcal{F}_h}^{\Sigma_1}$.*

Fusions of algebras have a close link with unions of theories, which we will exploit later.

Lemma 3.3 *If E_1, E_2 are two equational theories of signature Σ_1, Σ_2, respectively, and $\mathcal{F}$ is a fusion of a model of E_1 and a model of E_2, then $\mathcal{F}$ is a model of $E_1 \cup E_2$.*

A proof of this lemma can be found in [BT99]. The same work also proves that, in the presence of certain conditions, the test for satisfiability in a fusion of two algebras can be reduced to a "local" satisfiability test in each of the algebras.

Proposition 3.4 *Let $\mathcal{A}_1$ be a Σ_1-algebra, $\mathcal{A}_2$ be a Σ_2-algebra, and $\Sigma := \Sigma_1 \cap \Sigma_2$ such that the reducts ${\mathcal{A}_1}^{\Sigma}, {\mathcal{A}_2}^{\Sigma}$ are both free in the same Σ-variety over respective sets of generators Y_1, Y_2 having the same cardinality. Let φ_1, φ_2 be two arbitrary first-order formulae of signature Σ_1, Σ_2, respectively. 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 there is a fusion of $\mathcal{A}_1$ and $\mathcal{A}_2$ in which $\varphi_1 \wedge \varphi_2$ is satisfiable.*

Notice that the proposition does not require that the whole algebras be free but just their *reducts* to the common signature. In the following, however, we will be interested in countably generated free Σ_i-algebras ($i = 1, 2$) whose reducts to the common signature $\Sigma_1 \cap \Sigma_2$ are also free, in the same variety, and over a countably infinite set of generators.

In [BT99] we used the notion of *constructors* mentioned earlier to identify certain theories whose free algebras behaved as above. We generalize this notion in the next subsection.

3.1 Theories Admitting Constructors

We are interested in free models whose reducts to some shared subsignature 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 below.

In the following, Ω will be an at most countably infinite functional 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 3.5 (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.*

This definition is a proper generalization of the definition of constructors given in [BT99], which in addition 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 constructor for the theory E. Similarly, the empty signature is a set of constructor for E, as any model of E is free over its whole carrier in $E^{\emptyset}$, which is $\{v \equiv v \mid v \in V\}$. Also, the constant symbols of Ω are easily shown to be a set of constructors for E.

If E is axiomatized by the union of two theories E_1, E_2 of respective, *disjoint* signatures, Σ_1, Σ_2, then Σ_i $(i = 1, 2)$ is a set of constructors for E. This is not immediate but can be shown as a consequence of some results in [BS98].

At any rate, Definition 3.5 is a rather abstract formulation which may make it difficult to say for a given theory E and signature Σ whether is a set of constructors for E. A more concrete, syntactic characterization of theories admitting constructors is described below. 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. We will refer to these terms as the *G-variables* of t. 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 3.6 (Σ-base) *A subset G of $T(\Omega, V)$ 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 $s_1(\bar{r}_1), s_2(\bar{r}_2) \in T(\Sigma, G)$,*
$$s_1(\bar{r}_1) =_E s_2(\bar{r}_2) \quad \textit{iff} \quad s_1(\bar{v}_1) =_E s_2(\bar{v}_2),$$
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.

We will say that E *admits a Σ-base* if some subset G of $T(\Omega, V)$ is a Σ-base of E.

Theorem 3.7 (Characterization of constructors) *The signature Σ is a set of constructors for E iff E admits a Σ-base.*

Proof. Let $\mathcal{A}$ be an Ω-algebra free in E over some countably infinite set X, and α any bijective valuation of V onto X.[3]

($\Leftarrow$) Where G is any Σ-base of E, let

$$Y \quad := \quad \{[\![r]\!]^{\mathcal{A}}_{\alpha} \mid r \in G\}.$$

Since $V \subseteq G$ by definition of Σ-base, it is immediate that $X \subseteq Y$. We show that $\mathcal{A}^{\Sigma}$ is free in E^{Σ} over Y.

Let us start by observing that, since $\mathcal{A}$ is a model of E, its reduct $\mathcal{A}^{\Sigma}$ is a model of E^{Σ}. Next, we show that $\mathcal{A}^{\Sigma}$ is generated by Y. In fact, let a be an element of A—which is also the carrier of $\mathcal{A}^{\Sigma}$. We know that, as an Ω-algebra, $\mathcal{A}$ is generated by X; thus there exists a term $t \in T(\Omega, V)$ such that $a = [\![t]\!]^{\mathcal{A}}_{\alpha}$. By condition (2) of Definition 3.6, the term $t \in T(\Omega, V)$ is equivalent in E to a term $s(\bar{r}) \in T(\Sigma, G)$. Since $\mathcal{A}$ is a model of E, this implies that $a = [\![t]\!]^{\mathcal{A}}_{\alpha} = [\![s(\bar{r})]\!]^{\mathcal{A}}_{\alpha}$, from which it easily follows by definition of Y that a is Σ-generated by Y.

The above entails that $\mathcal{A}^{\Sigma}$ satisfies the first two conditions of Proposition 2.1. To show that it is free in E^{Σ} then it is enough to show that it also satisfies the third condition of the same proposition.

Thus, let $s_1(\bar{v}_1), s_2(\bar{v}_2) \in T(\Sigma, V)$ and assume that $\mathcal{A}^{\Sigma}, \alpha' \models s_1(\bar{v}_1) \equiv s_2(\bar{v}_2)$ for some injection α' of $V_0 := \mathcal{V}ar(s_1(\bar{v}_1) \equiv s_2(\bar{v}_2))$ into Y. By definition of Y we know that, for all $v \in V_0$, there is a term $r_v \in G$ such that $\alpha'(v) = [\![r_v]\!]^{\mathcal{A}}_{\alpha}$. Using these terms we can construct two tuples $\bar{r}_1$ and $\bar{r}_2$ of terms in G such that, for $i = 1, 2$, the term $s_i(\bar{r}_i)$ is obtained from $s_i(\bar{v}_i)$ by replacing each variable v in $\mathcal{V}ar(s_i(\bar{v}_i))$ by the term r_v, and $\mathcal{A}, \alpha \models s_1(\bar{r}_1) \equiv s_2(\bar{r}_2)$. Since $\mathcal{A}$ is free in E over X and α is injective as well we can conclude by Proposition 2.1(3) that $s_1(\bar{r}_1) =_E s_2(\bar{r}_2)$.

Because of the assumption that α' is injective, we know that $r_u \neq_E r_v$ for distinct variables $u, v \in V_0$. Thus, considered the other way around, the equation $s_1(\bar{v}_1) \equiv s_2(\bar{v}_2)$ can be obtained from $s_1(\bar{r}_1) \equiv s_2(\bar{r}_2)$ by abstracting the terms $\bar{r}_1, \bar{r}_2$ so that two terms are abstracted by the same variable iff they are equivalent in E. By Definition 3.6(3) then we obtain that $s_1(\bar{v}_1) =_E s_2(\bar{v}_2)$. Since the terms $s_1(\bar{v}_1), s_2(\bar{v}_2)$ are Σ-terms, this is the same as saying that $s_1(\bar{v}_1) =_{E^{\Sigma}} s_2(\bar{v}_2)$.

($\Rightarrow$) Now assume that Σ is a set of constructors for E, which implies that $\mathcal{A}^{\Sigma}$ is free in E^{Σ} over some set Y such that $X \subseteq Y$. First notice that, since $\mathcal{A}$ is generated by X, for every element y of Y there is a term r in $T(\Omega, V)$ such that $y = [\![r]\!]^{\mathcal{A}}_{\alpha}$. Then let

$$G \quad := \quad \{r \in T(\Omega, V) \mid [\![r]\!]^{\mathcal{A}}_{\alpha} \in Y\}.$$

[3] Such a valuation α exists since V is assumed to be countably infinite.

We show that G is a Σ-base of E.

Since $X \subseteq Y$, it is immediate that every $v \in V$ is in G, which means that G satisfies the first condition in Definition 3.6. The second condition easily follows from the fact that $\mathcal{A}^\Sigma$ is Σ-generated by Y. Similarly, the third condition follows from Proposition 2.1(3). $\square$

We will use sets such as the set Y defined in the proof above often enough to justify the following notation. If T is a subset of $T(\Omega, V)$, $\mathcal{A}$ an Ω-algebra free in E over a countably-infinite set X, and α a bijective valuation of V onto X we will denote by $[\![T]\!]^{\mathcal{A}}_\alpha$ the set $\{[\![t]\!]^{\mathcal{A}}_\alpha \mid t \in T\}$.

Corollary 3.8 *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 $Y := [\![G]\!]^{\mathcal{A}}_\alpha$, 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 questions 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 [BT99]. When E^Σ is not collapse-free, however, $\mathcal{A}^\Sigma$ may be free in it over more than one set of generators, not all of which include X.

For instance, consider the Ω-theory $E_1 := \{g(g(x)) = x\}$ and let $\Sigma := \Omega$. Let $\mathcal{A}$ be an Ω-algebra free in E_1 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 ${E_1}^\Sigma$ 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.

Going back to the notion of Σ-base, it should be clear that a theory E with constructors Σ usually admits many Σ-bases. For instance, if G is a Σ-base of E, any set G' equal to G modulo equivalence in E is also a Σ-base of E. In fact, we can say even more.

Proposition 3.9 *Let G be a Σ-base of E. Then, the following holds.*

- *For all bijective renamings π of V onto itself, the set $G' := \{\pi(r) \mid r \in G\}$ is also a Σ-base of E.*

- *For all sets G' such that $G \subseteq G' \subseteq T(\Omega, V)$, the set G' is a Σ-base of E iff G' coincides with G modulo equivalence in E.*

Notice that all the Σ-bases G' in the above result are really syntactical variants of G. In fact, let $\mathcal{A}$ be a free model of E over a countably infinite set X and α a bijection of V onto X. Then, for every G' above there is a bijection α' of V onto X such that $[\![G']\!]^{\mathcal{A}}_{\alpha'}$ coincides with $[\![G]\!]^{\mathcal{A}}_{\alpha}$, the set of free generators for the Σ-reduct of $\mathcal{A}$. If G is closed under bijective renaming or is equal to G' modulo E, one such α' is α itself; otherwise, if $G' = \{\pi(r) \mid r \in G\}$ with π as above, one α' is $\alpha \circ \pi^{-1}$.

Now, although an equational theory E with constructors Σ may have free models with Σ-reducts admitting distinct sets of free generators (as for instance the theory E_1 in the previous example), it is not clear whether it can have *substantially* different Σ-bases, i.e., Σ-bases yielding distinct sets of free generators.[4] For now, we only know that this is impossible if E^{Σ} is collapse-free. The reason is that then the Σ-reduct of the infinitely generated free model of E has exactly one set of free generators, and so all Σ-bases of E, if any, denote that unique set [BT99].

As it turns out, when E^{Σ} is collapse-free, E has a Σ-base $G_E(\Sigma, V)$ that is closed under bijective renaming and equivalence in E, and as such includes all the Σ-bases of E. In [BT99], where the definition of constructors included the collapse-freeness requirement on E^{Σ}, this maximal Σ-base was defined as follows:

$$G_E(\Sigma, V) \;:=\; \{r \in T(\Omega, V) \mid r \neq_E t \text{ for all } t \in T(\Omega, V) \text{ with } t(\epsilon) \in \Sigma\}. \quad (1)$$

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. In [BT99], it is shown that $G_E(\Sigma, V)$ is a Σ-base of E exactly when Σ is a set of (collapse-free) constructors.

To summarize, the following holds for $G_E(\Sigma, V)$.

Proposition 3.10 *Let $G_E(\Sigma, V)$ be the set defined in (1). 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.*

Normal Forms

According to Definition 3.6, if a set G is a Σ-base of an Ω-theory E, every Ω-term t is equivalent in E to a term $s(\bar{r}) \in T(\Sigma, G)$. We will call $s(\bar{r})$ *a G-normal form of t in E*.[5] We will say that a term t is *in G-normal form* if it is already of the

[4] The theory E_1 in the previous example does have two sets, V and $\{g(v) \mid v \in V\}$, that yield distinct sets of free generators for the reduct. However, only the former is a Σ-base of E_1; the latter is not because it does not satisfy Condition 1 of Definition 3.6.

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

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 speak just of *normal forms* instead of G-normal forms whenever the specific Σ-base G in question is clear from context or not relevant.

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

Definition 3.11 (Computable Normal Forms) *Let Σ be a set of constructors for the equational theory E over the signature Ω. 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 its G-variables, i.e., those terms $\bar{r}$ of G such that $t = s(\bar{r})$ for some Σ-term s. Now, in the combination procedure shown in Section 4, sometimes we will need to first compute the normal form of a term and then decompose this normal form 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.

Proposition 3.12 *Where Σ is a set of constructors for the equational theory E over the signature Ω, let G be a Σ-base of E and $t \in T(\Sigma, G)$. If G is recursive, there is an effective way of identifying a term $s(\bar{v}) \in T(\Sigma, V)$ and a sequence $\bar{r}$ of terms in G such that $t = s(\bar{r})$.*

Proof. Let $t \in T(\Sigma, G)$. Another consequence of the fact that the terms in G may start with a Σ symbol is that, in general, there may be more than one term $s(\bar{v})$ and tuple $\bar{r}$ such that $t = s(\bar{r})$. This proof by structural induction shows just one way to identify $s(\bar{v})$ and $\bar{r}$.

(Base Case) If $t \in V$ the claim is trivially true because $t \in G$ by definition of G.

(Inductive Step) Let t be the term $f(t_1, \ldots, t_n)$ with $f \in \Omega$. If t is in G, which we can effectively check because G is recursive, we can choose any $s \in V$ and let $\bar{r}$ be made of just t itself. If t is in not in G, then f must be a Σ-symbol since $t \in T(\Sigma, G)$ by assumption. For $j \in \{1, \ldots, n\}$, let $s_j(\bar{r}_j)$ be the decomposition of the term t_j, which is computable by induction. Let $f(s_1, \ldots, s_n)(\bar{v})$ be the term obtained from t by replacing with fresh variables $\bar{v}$ all the occurrences in t of the terms in $\bar{r}_1, \ldots, \bar{r}_n$ so that identical occurrences are replaced by the same variable. Where $\bar{r}$ consists, in order, of the terms of G abstracted by $\bar{v}$, it is immediate that $s(\bar{v}) = f(s_1, \ldots, s_n)(\bar{v}) \in T(\Sigma, V)$ and $t = s(\bar{r})$. □

Examples

We provide below some examples of equational theories admitting constructors in the sense of Definition 3.5. But first, let us consider some immediate counter-examples:

- The signature $\Sigma := \{\mathsf{s}\}$ is not a set of constructors for the theory E axiomatized by $\{x \equiv \mathsf{p}(\mathsf{s}(x)),\ x \equiv \mathsf{s}(\mathsf{p}(x))\}$. It is possible to show that, in constrast with the definition of constructors, the Σ-reduct of any free model of E over a countably infinite set is not itself free, because it does not admit a non-redundant[6] set of generators, a necessary condition for an algebra to be free.

- The signature $\Sigma := \{f\}$ is not a set of constructors for the theory E axiomatized by $\{g(x) \equiv f(g(x))\}$. In fact, since E^{Σ} is clearly collapse-free we know that any Σ-base of E, if any, is included in the set $G_E(\Sigma, V)$ defined earlier. But $G_E(\Sigma, V)$ is simply V in this case, and it is immediate that no subset of V satisfies condition 2 of Definition 3.6.

- Finally, the signature $\Sigma := \{f\}$ is not a set of constructors for theory E axiomatized by $\{f(g(x)) \equiv f(f(g(x)))\}$. Again, E^{Σ} is clearly collapse-free. Moreover, $G_E(\Sigma, V) = V \cup \{g(t) \mid t \in T(\Omega, V)\}$. It is easy to see that conditions (1) and (2) of Definition 3.6 hold for $G_E(\Sigma, V)$. However, condition (3) does not since $f(g(x)) =_E f(f(g(x)))$, although $f(y) \neq_E f(f(y))$.

Example 3.13 In [BT99] it is shown that:

- the signature $\Sigma := \{\mathsf{0}, \mathsf{s}\}$ is a set of constructors for the theory E_1 axiomatized by the equations:

$$\begin{array}{rcl} x + (y + z) & \equiv & (x + y) + z, \\ x + y & \equiv & y + x, \\ x + \mathsf{s}(y) & \equiv & \mathsf{s}(x + y), \\ x + \mathsf{0} & \equiv & x; \end{array}$$

- the signature $\Sigma := \{\mathsf{0}, \mathsf{s}\}$ is a set of constructors for the theory E_2 axiomatized by the equations:

$$\begin{array}{rcl} \mathsf{mod2}(\mathsf{0}) & \equiv & \mathsf{0}, \\ \mathsf{mod2}(\mathsf{s}(\mathsf{0})) & \equiv & \mathsf{s}(\mathsf{0}), \\ \mathsf{mod2}(\mathsf{s}(\mathsf{s}(x))) & \equiv & \mathsf{mod2}(x), \\ \mathsf{mod2}(\mathsf{mod2}(x)) & \equiv & \mathsf{mod2}(x); \end{array}$$

[6] A set of generators for an algebra $\mathcal{A}$ is *redundant* if one of its proper subsets also generates $\mathcal{A}$.

- the signature $\Sigma := \{\mathsf{0}, \mathsf{1}, \cdot\}$ is a set of constructors for the theory E_3 axiomatized by the equations:

$$\begin{array}{rcl} x \cdot (y \cdot z) & \equiv & (x \cdot y) \cdot z, \\ \mathsf{rev}(\mathsf{0}) & \equiv & \mathsf{0}, \\ \mathsf{rev}(\mathsf{1}) & \equiv & \mathsf{1}, \\ \mathsf{rev}(x \cdot y) & \equiv & \mathsf{rev}(y) \cdot \mathsf{rev}(x), \\ \mathsf{rev}(\mathsf{rev}(x)) & \equiv & x. \end{array}$$

In the example above, the restriction of each theory to the constructor symbols is collapse-free. That is not the case for the theory in the next example.

Example 3.14 Consider the signature $\Omega := \{\mathsf{0}, \mathsf{p}, \mathsf{s}, -\}$ and the equational theory E axiomatized by the equations:

$$\begin{array}{rcl} \mathsf{s}(\mathsf{p}(x)) & \equiv & x \\ \mathsf{p}(\mathsf{s}(x)) & \equiv & x \\ -\mathsf{0} & \equiv & \mathsf{0}, \\ -\mathsf{s}(x) & \equiv & \mathsf{p}(-x), \\ -\mathsf{p}(x) & \equiv & \mathsf{s}(-x). \end{array}$$

The signature $\Sigma := \{\mathsf{0}, \mathsf{p}, \mathsf{s}\}$ is a set of constructors for E. To prove it we show that the set $G := V \cup \{-v \mid v \in V\}$ is a Σ-base of E.

Clearly, $V \subseteq G$. To show the remaining two conditions of Definition 3.6, note that orienting the axioms above from left to right produces a confluent and terminating rewrite system R. Thus, two terms are equal modulo E iff their R-normal forms are syntactically identical.

Now, Condition 2 of Definition 3.6 is satisfied since, given an Ω-term, its R-normal form is in $T(\Sigma, G)$. This is an immediate consequence of the fact that (because of the last three rules of R) any term containing the minus symbol in front of $\mathsf{0}$, p, or s is R-reducible. Therefore, in R-normal forms, minus can only occur in front of variables.

All we need to show then is that Condition 3 of Definition 3.6 is also satisfied. Thus, let $s_1(\bar{r}_1), s_2(\bar{r}_2)$ be terms in $T(\Sigma, G)$ such that $s_1(\bar{r}_1) =_E s_2(\bar{r}_2)$. Since R is confluent and terminating, there exists a term t such that $s_1(\bar{r}_1) \stackrel{*}{\rightarrow}_R t$ and $s_2(\bar{r}_2) \stackrel{*}{\rightarrow}_R t$. Since in the terms $s_1(\bar{r}_1), s_2(\bar{r}_2)$ (as well as in any term occurring in the reduction chains) the minus symbol can only occur in front of variables, the reduction chains make use of the first two rules of R only. Consequently, $s_1(\bar{r}_1)$ and $s_2(\bar{r}_2)$ are equal modulo the first two axioms of E. Given that these axioms do not contain the minus symbol, it is easy to see that this implies that $s_1(\bar{v}_1) =_E s_2(\bar{v}_2)$. Since the other direction of the bi-implication of Condition 3 is trivial, this completes the proof that $G := V \cup \{-v \mid v \in V\}$ 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 following subsection, however, we would like to point out another, perhaps less obvious, class of examples for which our combination approach could provide fresh insights and results.

3.2 Constructors and Modal Logics

For all normal modal logics, equivalence of formulae is a congruence relation on formulae that is closed under substitution (see [Gol76] or Chapter 4 in [Kra99]). For example, consider the basic modal logic K [Fit93]. Here, the signature Σ_{K} contains the Boolean operators ($\wedge$, $\vee$, $\neg$), the Boolean constant $\top$ (for truth), and the unary (modal) operator $\Box$.[7] Equivalence of formulae in K can be axiomatized [Lem66] 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 formulae in K is decidable iff the word problem for E_{K} is decidable. For example, ϕ is valid iff $\phi =_{E_{\mathsf{K}}} \top$. Since satisfiability in K is indeed decidable[8] the word problem for E_{K} is also decidable.

The problem of combining modal logics has been thoroughly investigated (see, e.g., [Hem94, KW97]). In particular, there are very general results on how decidability of the component logics transfers to their combination (called *fusion* or *join* 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 can be shown 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 in the modal logic context. For example, let Σ be the subsignature of Σ_{K} that consists of $\wedge$, $\vee$, $\neg$, and $\top$. It is easy to see 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

[7] We do not explicitly introduce the diamond operator since it can be expressed using $\Box$ and $\neg$.
[8] In fact, it is a well-known PSPACE-complete problem.

algebra, and since all countably infinite atomless Boolean algebras are known to be isomorphic [Kop88], 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 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 sublanguage of K. Such a sublanguage, which is not Boolean closed, is particularly interesting because the current combination results in modal logic are restricted to Boolean closed languages.

Example 3.15 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.

It is not hard to show [BN98][9] that equivalence of such formulae is axiomatized by the theory E^0_{K}, which consists of the axioms

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

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 [BN98] and the characterization of equivalence[10] proved in the same paper. The concept-based normal form of a formula is obtained by exhaustively applying the rewrite rules

$$\Box(x \wedge y) \to \Box(x) \wedge \Box(y), \ \Box(\top) \to \top, \ x \wedge \top \to x, \ \top \wedge x \to x.$$

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 this shows that all prerequisites for our combination approach to apply to E^0_{K} are satisfied.

Interestingly, if we consider the conjunctive fragment of the modal logic S_4 [Fit93] in place of K, we obtain a quite different behavior. 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).

[9] Note that [BN98] employs description logic syntax rather than modal logic syntax for formulae.
[10] This characterization also shows that the word problem for E^0_{K} is decidable in polynomial time.

Example 3.16 As an easy consequence of the usual axiomatization of S_4 we can show that equivalence in the conjunctive fragment of S_4 is axiomatized by the equational theory

$$E^0_{\mathsf{S}_4} \;:=\; E^0_{\mathsf{K}} \cup \{\Box(x) \wedge x \equiv \Box(x),\ \Box(\Box(x)) \equiv \Box(x)\}.$$

We can show that $E^0_{\mathsf{S}_4}$ does not have a Σ^0-base. Intuitively, the reason is the following. Assume that G is a Σ^0-base. For any $v \in V$, both v and $\Box(v)$ (more precisely, at least one element of their $E^0_{\mathsf{S}_4}$-equivalence class) must belong to G. In fact, any term equivalent to v must contain v in its top level conjunction, and any term equivalent to $\Box(v)$ must contain a term of the form $\Box^n(v)$ for $n \geq 1$ in its top level conjunction. However, because of the axiom $\Box(x) \wedge x \equiv \Box(x)$, no set containing both v and $\Box(v)$ can satisfy the third condition in the definition of Σ-base.

3.3 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, and
- ${E_1}^{\Sigma} = {E_2}^{\Sigma}$.

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 $Y_i := [\![G_i]\!]^{\mathcal{A}_i}_{\alpha_i}$ where G_i is any Σ-base of E_i and α_i is any bijective valuation of V onto X_i.

Proposition 3.17 *Let φ_1, φ_2 be two first-order formulae of respective signatures Σ_1 and 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 $E_1 \cup E_2$.*

Proof. Let $E_0 := {E_1}^{\Sigma}$ $(= {E_2}^{\Sigma})$. By Corollary 3.8, ${\mathcal{A}_i}^{\Sigma}$ is free in E_0 over Y_i for $i = 1, 2$. Moreover, Y_1 and Y_2 have the same cardinality because, for $i = 1, 2$, $X_i \subseteq Y_i \subseteq A_i$ by construction of Y_i, and X_i and A_i are countably infinite by assumption. By Proposition 3.4 then $\varphi_1 \wedge \varphi_2$ is satisfiable in a fusion of $\mathcal{A}_1$ and $\mathcal{A}_2$, which is a model of $E_1 \cup E_2$ by Lemma 3.3. $\Box$

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 next two sections, we will see 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. Section 4 shows that, 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. Section 5 shows that the property of being a set of constructors is itself modular.

4 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 a non-trivial equational theory with decidable word problem. Such a procedure will be obtained as a 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 $i = 1, 2$:

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

Later we will also assume that ${E_1}^{\Sigma} = {E_2}^{\Sigma}$. Such a restriction is not required to show the termination and soundness properties of the combination procedure. It will be used only to prove the procedure's completeness.

To decide the word problem for $E := E_1 \cup E_2$, we consider the satisfiability problem for quantifier-free formulae of the form $s_0 \not\equiv t_0$, where s_0 and t_0 are terms in the signature of E. The first step of our procedure transforms a formula of this form into a conjunction of pure formulae by means of variable abstraction. To define the purification process in more detail, we need to introduce a little more notation and some new concepts.

4.1 Abstraction Systems

We will often use finite sets of formulae in place of conjunctions of such formulae, that is, we will treat a finite set S of formulae as the formula $\bigwedge_{\varphi \in S} \varphi$. We will then say that S is satisfiable in a theory iff the conjunction of its elements is satisfiable in that theory.

We can define a procedure which, given a disequation $s_0 \not\equiv t_0$ with $s_0, t_0 \in T(\Sigma_1 \cup \Sigma_2, V)$, produces a set $AS(s_0 \not\equiv t_0)$ consisting of pure equations and disequations 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 *purification procedure* starts with the set $S_0 := \{x \not\equiv y, x \equiv s_0, y \equiv t_0\}$, where x, y are distinct variables not occurring in s_0, t_0, if s_0 and t_0 are not variables. If s_0 (t_0) is a variable, the procedure uses s_0 in place of x (t_0 in place of y), and omits the corresponding (trivial) equation. Assume that a finite set S_i consisting of $x \not\equiv y$ and equations of the form $u \equiv s$ (where $u \in V$ and $s \in T(\Sigma_1 \cup \Sigma_2, V) \setminus V$) has already been constructed. If S_i contains an equation $u \equiv s$ such that s has an alien subterm t at position p, then S_{i+1} is obtained from S_i by replacing $u \equiv s$ by the equations $u \equiv s'$ and $v \equiv t$, where v is a variable not occurring in S_i, and s' is obtained from s by replacing t at position p by v. Otherwise, if all terms occurring in S_i are pure, the procedure stops and returns S_i.

It is easy to see that this process terminates and yields a set $AS(s_0 \not\equiv t_0)$ which is satisfiable in E iff $s_0 \not\equiv t_0$ is satisfiable in E. The set $AS(s_0 \not\equiv t_0)$ satisfies additional properties (see Proposition 4.3 below), whose importance will become clear later on.

Definition 4.1 *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 for all $u \equiv s, v \equiv t \in T$:*

$$(u \equiv s) \prec (v \equiv t) \quad \textit{iff} \quad v \in \mathcal{V}ar(s).$$

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 $(v \equiv t) \prec^+ (v \equiv t)$.*

Notice that, when $\prec$ is acyclic, $\prec^*$ is a partial order, and $\prec^+$ is the corresponding strict partial order.

Definition 4.2 (Abstraction System) *The 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 $t \notin T(\Sigma_i, V)$.

Condition (1) above states that T consists of equations between variables and pure non-variable terms; 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 $\Sigma_2 \setminus \Sigma$.

We will call the variables occurring in an abstraction system S as the left-hand side of an equation the *left-hand side variables* of S. Similarly, we will call the terms occurring in an abstraction system S as the right-hand side of an equation the *right-hand side terms* of S.

The following proposition is an easy consequence of the definition of the purification procedure.

Proposition 4.3 *The set $S := AS(s_0 \not\equiv t_0)$ obtained by applying the purification procedure to the disequation $s_0 \not\equiv t_0$ is an abstraction system. Furthermore, $\exists\bar{v}.S \leftrightarrow (s_0 \not\equiv t_0)$ is logically valid, where $\bar{v}$ are all the left-hand side variables of S.*

In particular, the second part of the proposition implies that a disequation $s_0 \not\equiv t_0$ is satisfiable in E iff $AS(s_0 \not\equiv t_0)$ is satisfiable in E. Note, however, that the statement in the proposition is considerably stronger: if $\mathcal{A}$ is a $(\Sigma_1 \cup \Sigma_2)$-algebra and α a valuation that satisfies $s_0 \not\equiv t_0$ in $\mathcal{A}$, then there exists a valuation α' that coincides with α on $\mathcal{V}ar(s_0 \not\equiv t_0)$ and satisfies $AS(s_0 \not\equiv t_0)$, and vice versa. In fact, the left-hand side variables in $AS(s_0 \not\equiv t_0)$ are fresh variables that do not occur in $s_0 \not\equiv t_0$, and all the newly introduced variables are left-hand side variables. Thus, the variables in $\mathcal{V}ar(s_0 \not\equiv t_0)$ are the free variables of both $s_0 \not\equiv t_0$ and $\exists\bar{v}.S$, which means that they are (implicitly) universally quantified on the outside in the equivalence $\exists\bar{v}.S \leftrightarrow (s_0 \not\equiv t_0)$. We will appeal to this stronger statement in Section 5.

Abstraction Systems as Directed Acyclic Graphs

Any abstraction system $\{x \neq y\} \cup T$ induces a graph $\mathcal{G}$ whose set of *nodes* is T and whose set of *edges* consists of all the pairs $(a_1, a_2) \in T \times T$ such that $a_1 \prec a_2$. According to Definition 4.2, $\mathcal{G}$ is in fact a directed acyclic graph (or *dag*).[11] For

[11] Observe that $\mathcal{G}$ need not be a tree or even be connected.

notational convenience, we will sometimes identify an abstraction system with the graph induced by it.

Assuming the standard definition of path between two nodes and of length of a path in a dag, we define below a notion of *height* of a node, which measures the longest possible path from a "root" of the graph to the node. This notion will be used in the definition of our combination procedure, and it will be important for the termination proof.

Definition 4.4 (Node Height) *Let $\mathcal{G} := (\mathsf{N}, \mathsf{E})$ be a dag with finite sets of nodes and edges. A node $a \in \mathsf{N}$ is a* root *of $\mathcal{G}$ iff there is no $a' \in \mathsf{N}$ such that $(a', a) \in \mathsf{E}$.*[12] *The function $\mathsf{h} \colon \mathsf{N} \longrightarrow \mathbb{N}$ is defined as follows. For all $a \in \mathsf{N}$,*

- $\mathsf{h}(a) = 0$*, if a is a root of $\mathcal{G}$;*
- $\mathsf{h}(a)$ *equals the maximum of the lengths of all the paths from the roots of $\mathcal{G}$ to a, otherwise.*[13]

Later, we will appeal to the following easily provable facts about the height function introduced above.

Lemma 4.5 *The following holds for every finite dag $\mathcal{G}$ and associated height function h.*

1. *For all nodes a, b of $\mathcal{G}$, if there is a non-empty path from a to b then $\mathsf{h}(a) < \mathsf{h}(b)$.*
2. *Adding an edge from a node of $\mathcal{G}$ to another of greater height does not change the height of any node of $\mathcal{G}$.*
3. *Removing an edge in $\mathcal{G}$ does not increase the height of any node of $\mathcal{G}$ (although it may decrease the height of some).*
4. *Removing a node and relative edges from $\mathcal{G}$ does not increase the height of the remaining nodes (although it may decrease the height of some).*

We say that an equation of an abstraction system is *reducible* iff its right-hand side is neither in G_1 nor in G_2 (the respective Σ-bases of E_1 and E_2).

Definition 4.6 (Node Reducibility) *Let $(T, \prec)$ be the dag induced by the abstraction system $\{x \neq y\} \cup T$ and let $a \in T$. We say that the* reducibility *of a is 1, and write $\mathsf{r}_A(a) = 1$, if a is reducible; we say that it is 0, and write $\mathsf{r}_A(a) = 0$, otherwise.*

[12] Because of the acyclicity condition, any finite dag has at least one root.

[13] This maximum exists because $\mathcal{G}$ is finite and acyclic.

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.

Figure 1: The Combination Procedure.

4.2 The Combination Procedure

In Section 3.1, 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-variable 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 $G_i \setminus V$.

Figure 1 describes a procedure that decides the word problem for the theory $E := E_1 \cup E_2$ by deciding, as we will show, 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. The procedure applies a number of transformation rules to a certain abstraction system until no more rules apply.

The main idea of the procedure is to see whether the disequation between the two input terms is satisfiable in E by turning the disequation into an abstraction system, and then propagating some of the equations between variables that are valid in one of the single theories. The transformations the initial system goes through will eventually produce an abstraction system whose initial formula has the form $v \not\equiv v$ iff the initial disequation $s_0 \not\equiv t_0$ is unsatisfiable in E (that is, iff $s_0 =_E t_0$).

During the execution of the procedure, the abstraction system S on which the procedure works is repeatedly modified by the application of one of the derivation rules defined in Figure 2 and Figure 3. We 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. It is understood that any two formulae explicitly listed in the premise of a rule are distinct.

The transformations applied by the procedure to the set S above boil down to one of two operations:[14]

[14]The formal description of the individual transformations is actually somewhat more complex,

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

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

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

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

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

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

$$\textbf{Simpl}\ \frac{T \quad x \equiv t}{T}$$

if $x \notin \mathcal{V}ar(T)$.

Figure 2: The Derivation Rules.

- replace a variable in S by another one in S;
- replace a variable in S by a shared term.

This kind of replacement implements a form of constraint propagation between the decision procedures for the word problem in the component theories E_1 and E_2. The main part of the correctness proof for the combination procedure will be to show that such a restricted form of constraint propagation is enough for our purposes.

Coming to the single derivation rules, **Coll1** and **Coll2** basically remove from S collapse equations that are valid in E_1 or E_2, while **Ident** identifies any two variables

but for technical reasons.

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

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) $\emptyset \subsetneq \bar{r} \subseteq G_i \setminus T(\Sigma, V)$,
(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 $(y \equiv r) \in T$.

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

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 $(y \equiv r) \in T$.

Figure 3: More Derivation Rules.

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 associated to 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.[15]

Simpl eliminates those equations that have become unreachable along a $\prec$-path from the initial disequation because of the application of previous rules. As we will see, this rule is not essential but it reduces clutter in S by eliminating equations that do not contribute to the solution of the problem anymore. It can be used to obtain

[15]Notice that other authors, especially in programming languages theory, would denote the same substitution by $T[t/x]$ instead. We prefer our convention because we find it more intuitive, especially in the case of composed substitutions.

optimized, complete implementations of the combination procedure.

The derivation rules **Shar1** and **Shar2** are shown separately because they apply only if Σ_1 and Σ_2 are non-disjoint. They are used to propagate the constraint information represented by shared terms. In both, the main idea is to push shared function symbols towards lower positions of the $\prec$-chains they belong to so that they can be processed by other rules. 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 into all equations whose right-hand sides contain x. The exact formulation of the rules is somewhat more complex since we must ensure that the rules do not apply repeatedly to the same equation and the resulting system is again an abstraction system. In particular, the rules must preserve the "alternating signature" condition (3b) of Definition 4.2.

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 $j \in \{1, \ldots, n\}$. Observe that 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; e.g., $[\bar{z}/\bar{r}]$ replaces the variable z_j by r_j for each $j \in \{1, \ldots, n\}$.

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,[16] each variable in V_0 is fresh with respect to the current set S. Such an assumption can be made without loss of generality. In fact, since each G_i is closed under renaming by assumption, applying any injective renaming of $\mathcal{V}ar(NF^{\Sigma}_{E_i}(t))$ to $NF^{\Sigma}_{E_i}(t)$ yields a term still in $T(\Sigma, G_i)$. In particular, we can choose a renaming that fixes the variables in $\mathcal{V}ar(t)$ and moves those in V_0 to fresh variables. This process is clearly effective and yields a term also equivalent to t in E_i.

In both **Shar** rules it is required that the normal form of t be a non-variable term. The reason for this restriction is that the rules **Coll1** and **Coll2** already take care of the case in which a Σ_i-term is equivalent in E_i to a variable. 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 4.2(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 $x \equiv s$, which is both a Σ_1- and a Σ_2-

[16] This might happen because Definition 3.11 does not necessarily entail that all the variables of $NF^{\Sigma}_{E_i}(t)$ occur in t.

equation, may inherit some of these successors from $x \equiv t$.[17] **Shar2** restores this property by merging into $x \equiv s$ all of its immediate successors—which are collected, if any, in the set $\bar{y}_1 \equiv \bar{r}_1$ thanks to Condition (e) in the rule. The replacement of $\bar{y}_1$ by $\bar{r}_1$ in **Shar1** is done for similar reasons. In **Shar2**, the restriction that all 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, as we will see.

We prove below that the combination procedure decides the word problem for $E = E_1 \cup E_2$ by showing that it terminates on all inputs, is sound and, whenever ${E_1}^\Sigma = {E_2}^\Sigma$, is also complete.

4.3 The Correctness Proof

In this subsection, we will consider a countable family $\mathcal{S} := \{S_j \mid j \geq 0\}$ such that S_0 is an abstraction system and for all $j > 0$, S_j is either identical to S_{j-1} or is derived from S_{j-1} by an application of the rule **Coll1**, **Coll2**, **Simpl**, **Ident**, **Shar1**, or **Shar2**.

In particular, $\mathcal{S}$ may correspond to the family generated by one execution of the combination procedure, where S_0 is the abstraction system $AS(s_0 \not\equiv t_0)$ obtained by applying the purification procedure to the input disequation, and $S_j := S_n$ for all $j > n$ if Step 2 of the combination procedure is executed only n times. In general, however, the first element of $\mathcal{S}$ may be an arbitrary abstraction system.

For all $j > 0$, we will denote by $\prec_j$ the relation $\prec$ on the equational part of S_j (cf. Definition 4.1).

We start by showing that all the elements of $\mathcal{S}$ are in fact abstraction systems. The proof of acyclicity (Condition 2 in Definition 4.2) will be facilitated by the following lemma, whose simple proof is omitted.

Lemma 4.7 *Let $<$ be a binary relation on a finite set A, and $a, b \in A$ be such that $b \not<^* a$. We denote the restriction of $<$ to $A \setminus \{a\}$ by $<_a$,[18] and consider the relations*

$$\begin{aligned} <_1 &:= <_a \cup \{\langle d, e\rangle \mid d < a, b < e\} \\ <_2 &:= <_a \cup \{\langle d, b\rangle \mid d < a\}. \end{aligned}$$

If $<$ is acyclic, then $<_1$ and $<_2$ are acyclic as well.

Lemma 4.8 *S_j is an abstraction system for all $j \geq 0$.*

[17] 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 $x \equiv t$.

[18] That is, $<_a := < \cap (A \setminus \{a\})^2$.

Proof. We prove the claim by induction on j. The induction base ($j = 0$) is immediate by assumption. Thus, assuming that $j > 0$ and that S_{j-1} is an abstraction system, consider the following cases, labeled by the derivation rule applied to S_{j-1} to obtain S_j.

Coll1. By the rule's definition, S_{j-1} and S_j must have the following form:

$$\begin{array}{lcllll} S_{j-1} & = & \{u \not\equiv v\} & \cup\ \{x \equiv t[y]\} & \cup\ \{y \equiv r\} & \cup\ T \\ S_j & = & \{u \not\equiv v\}[x/y] & \cup & \ \{y \equiv r\} & \cup\ T[x/r] \end{array}$$

Let $u' \not\equiv v' := (u \not\equiv v)[x/y]$. We show that S_j is an abstraction system with disequation $u' \not\equiv v'$.

If we take $\prec_{j-1}$ to be the relation $<$ of Lemma 4.7, $x \equiv t$ to be a, and $y \equiv r$ to be b, it is easy to see that $a < b$ and $\prec_j$ coincides with $<_1$ (as defined in the lemma). Now, $<$ is acyclic by induction and $b \not<^* a$ because $a < b$. By Lemma 4.7 then, $\prec_j$ is acyclic. This shows that condition (2) of Definition 4.2 holds.

Since applying the substitution $[x/r]$ does not change the left-hand sides of equations in T, it is immediate that condition (3a) of Definition 4.2 holds as well.

Finally, observe that x can appear in T only in an equation of the form $z \equiv s[x]$ and that $(z \equiv s) \prec_{j-1} (x \equiv t) \prec_{j-1} (y \equiv r)$. By induction, we know that there is an $i \in \{1, 2\}$ such that s and r are both in $T(\Sigma_i, V) \setminus T(\Sigma, V)$; therefore, the replacement of x by r in T occurs only inside terms in $T(\Sigma_i, V) \setminus T(\Sigma, V)$ and produces terms still in $T(\Sigma_i, V) \setminus T(\Sigma, V)$. It follows that S_j satisfies both conditions (1) and (3b) of Definition 4.2.

Coll2. The proof is essentially a special case of the one above, with r replaced by y. The proof of condition (2) of Definition 4.2 is, however, easier in this case. If we take $x \equiv t$ to be a and $\prec_{j-1}$ to be the relation $<$, then $\prec_j$ coincides with $<_a$ as defined in Lemma 4.7. If $<$ is acyclic, then its subrelation $<_a$ is acyclic as well.

Ident. By the rule's definition, S_{j-1} and S_j must have the following form:

$$\begin{array}{lclll} S_{j-1} & = & T \cup \{u \not\equiv v\} & \cup\ \{x \equiv s\} & \cup\ \{y \equiv t\} \\ S_j & = & (T \cup \{u \not\equiv v\})[x/y] & \cup & \ \{y \equiv t\}, \end{array}$$

Moreover, it is *not* the case that $(y \equiv t) \prec^+_{j-1} (x \equiv s)$, otherwise we would have that $\mathsf{h}(y \equiv t) < \mathsf{h}(x \equiv s)$. It is not difficult to see that this time $\prec_j$ is derivable from $\prec_{j-1}$ in the same way $<_2$ is derivable from $<$ in Lemma 4.7, where $x \equiv s$ is a and $y \equiv t$ is b. Again, the preconditions of the lemma are satisfied, and it follows that $\prec_j$ satisfies condition (2) of Definition 4.2. By induction, we know that x appears as the left-hand side of no equations in T, and so it is immediate that S_j satisfies condition (3a). It is also immediate that S_j satisfies condition (1).

Finally, to see that S_j also satisfies condition (3b), notice that T is unchanged if the height of $y \equiv t$ in S_{j-1} is zero. The reason is that, in this case, the height of $x \equiv s$ is also zero, which means that x does not occur in T. If $\mathsf{h}(y \equiv t) > 0$ and x

occurs in T, both s and t are elements of $T(\Sigma_i, V) \setminus T(\Sigma, V)$. But then we can argue that condition (3b) holds for S_j exactly as we did in the case of **Coll1**. It follows that S_j is an abstraction system with disequation $(u \not\equiv v)[x/y]$.

Simpl. Immediate consequence of the easily provable fact that, if $\{u \not\equiv v\} \cup T'$ is an abstraction system, then $\{u \not\equiv v\} \cup T$ is also an abstraction system for every $T \subseteq T'$.

Shar1. We know that S_{j-1} and S_j have the following form:

$$\begin{array}{lclcccccccc} S_{j-1} & = & T & & & \cup & \{u \not\equiv v\} & \cup & \{x \equiv t\} & \cup & \{\bar{y}_1 \equiv \bar{r}_1\} \\ S_j & = & T[x/s(\bar{y},\bar{z})[\bar{y}_1/\bar{r}_1]] & \cup & \{\bar{z} \equiv \bar{r}\} & \cup & \{u \not\equiv v\} & \cup & \{x \equiv s(\bar{y},\bar{r})\} & \cup & \{\bar{y}_1 \equiv \bar{r}_1\} \end{array}$$

To see that S_j satisfies Condition (1) of Definition 4.2, first notice that $s(\bar{y},\bar{r})$ is not a variable by precondition (c) of the rule, and that the terms in $\bar{r}$ are also non-variable terms. Because S_{j-1} is assumed to be an abstraction system, it satisfies the alternating signature assumption, and thus the terms in $\bar{r}_1$ are Σ_ι-terms with $\iota \in \{1,2\} \setminus \{i\}$. Since $s(\bar{y},\bar{z})$ is a Σ-term, we know that $s(\bar{y},\bar{z})[\bar{y}_1/\bar{r}_1]$ is also a Σ_ι-term. The alternating signature assumption for S_{j-1} also implies that any term in T containing x is a Σ_ι-term, and so the replacement of x by $s(\bar{y},\bar{z})[\bar{y}_1/\bar{r}_1]$ does not generate mixed terms.

Condition (3a) is satisfied because $\bar{z}$ consists of fresh variables with no repetitions. Condition (3b) is satisfied because

- every right-hand side $t'[x]$ of T, which is a term in $T(\Sigma_\iota, V) \setminus T(\Sigma, V)$ by induction hypothesis (cf. observation after Definition 4.2), is replaced by the term $t'[x/s(\bar{y},\bar{z})[\bar{y}_1/\bar{r}_1]]$, which is also in $T(\Sigma_\iota, V) \setminus T(\Sigma, V)$ by the above;
- the elements of $\bar{r}$ are not Σ-terms, have the same signature as t, and every immediate $\prec$-predecessor of an equation in $\bar{z} \equiv \bar{r}$ has the signature of the immediate predecessors of $x \equiv t$ in S_{j-1};
- all the immediate successors of $x \equiv s(\bar{y},\bar{r})$ are inherited from $x \equiv t$ because, by our assumptions on the variables of normal forms, the variables in $\mathcal{V}ar(s(\bar{y},\bar{r})) \setminus \mathcal{V}ar(t)$ do not occur in S_{j-1} (and without loss of generality also not in $\bar{z}$);
- $s(\bar{y},\bar{r})$ is not a Σ-term because the tuple $\bar{r}$ is non-empty and made of non-Σ-terms;
- if an equation $x' \equiv t'[x]$ in T is replaced by $x' \equiv t'[s(\bar{y},\bar{z})[\bar{y}_1/\bar{r}_1]]$, then any new successor of such an equation is an equation in $\bar{z} \equiv \bar{r}$ or a successor of an equation in $\bar{y}_1 \equiv \bar{r}_1$.

To show that Condition (2) is satisfied, we first prove that $T_j := S_j \setminus \{\bar{z} \equiv \bar{r}\}$ gives rise to an acyclic graph. This graph has essentially the same nodes (i.e., equations) as

S_{j-1}, although the right-hand sides of the equations may have changed. Even if there are possibly new edges, it is easy to see that there are no new connections between nodes, since any connection achieved by such a new edge in T_j can be achieved by a path in S_{j-1}. Since S_{j-1} induces an acyclic graph by assumption, this implies that the graph corresponding to T_j is acyclic as well. The additional nodes in S_j (i.e., the equations in $\bar{z} \equiv \bar{r}$) cannot cause a cycle either since any path through one of these nodes comes from a predecessor of $x \equiv t[\bar{y}]$ in S_{j-1} and goes to a successor of $x \equiv t[\bar{y}]$ in S_{j-1}. Thus, the cycle would have already been present in S_{j-1}.

Shar2. We know that S_{j-1} and S_j have the following form:

$$\begin{array}{lclcccccc} S_{j-1} & = & T & \cup & \{u \not\equiv v\} & \cup & \{x \equiv t[\bar{y}_1]\} & \cup & \{\bar{y}_1 \equiv \bar{r}_1\} \\ S_j & = & T[x/s[\bar{y}_1/\bar{r}_1]] & \cup & \{u \not\equiv v\} & \cup & \{x \equiv s[\bar{y}_1/\bar{r}_1]\} & \cup & \{\bar{y}_1 \equiv \bar{r}_1\} \end{array}$$

We can show that S_j satisfies Conditions (1), (2), (3a), and (3b) of Definition 4.2 essentially in the same way as in the **Shar1** case. For Condition (3b), additionally observe that we cannot use $x \equiv s$ in S_j because s is a shared term. By using $x \equiv s[\bar{y}_1/\bar{r}_1]$ instead, where the terms of $\bar{r}_1$ are non-shared by induction, we make sure that any successors of this equation is a successor of an equation in $\bar{y}_1 \equiv \bar{r}_1$. Since every equation in $\bar{y}_1 \equiv \bar{r}_1$ is a successor of $x \equiv t$ in S_{j-1},[19] and S_{j-1} satisfies Condition (3b) by induction, all the equations in $\bar{y}_1 \equiv \bar{r}_1$ have the same signature, which is also the signature of $x \equiv s[\bar{y}_1/\bar{r}_1]$. Thus, Condition (3b) for $x \equiv s[\bar{y}_1/\bar{r}_1]$ and its successors in S_j is satisfied since it is satisfied for the equations in $\bar{y}_1 \equiv \bar{r}_1$ and their successors in S_{j-1}. If the tuple $\bar{y}_1$ is empty, then $s[\bar{y}_1/\bar{r}_1] = s$ is a shared term, but this is not a problem since in this case the equation $x \equiv s$ does not have any predecessors or successors in S_j. □

The next result we prove about the combination procedure is that it halts on all inputs. For that we will make use of a well-founded ordering[20] on abstraction systems, defined in the following.

Let $>_l$ denote the lexicographic ordering over the set $P := \mathbb{N} \times \{0, 1\}$ obtained from the standard strict ordering over $\mathbb{N}$ and its restriction to $\{0, 1\}$. Where $\mathcal{M}(P)$ denotes the set of all finite multisets of elements of P, we will denote by $\sqsupset$ the multiset ordering induced by $>_l$. Intuitively, this ordering says that a multiset M is reduced by removing one or more elements from M, and replacing them by a finite number of $>_l$-smaller elements. It is possible to show that $\sqsupset$ is a well-founded total ordering on $\mathcal{M}(P)$ (see [DM79] for more details). As customary, we will denote by $\sqsupseteq$ the reflexive closure of $\sqsupset$.

Now, let h_j and r_j be the height and reducibility functions on the nodes of the dag induced by the abstraction system S_j, for $j \geq 0$. These functions can be used to

[19] Recall again that the variables in $\mathcal{V}ar(s) \setminus \mathcal{V}ar(t)$ do not occur in S_{j-1}.

[20] A strict ordering $>$ is well-founded if there are no infinitely decreasing chains $a_1 > a_2 > a_3 > \cdots$.

associate a finite multiset to S_j: the multiset M_j consisting of the pairs $(\mathsf{h}_j(a), \mathsf{r}_j(a))$ for every equation a in S_j. Notice that M_j is indeed a multiset: if S_j contains m irreducible nodes with height n, M_j contains m occurrences of the pair $(n, 0)$. Similarly, if S_j contains m reducible nodes with height n, M_j contains m occurrences of the pair $(n, 1)$.

The next lemma shows that each application of a derivation rule decreases the multiset associated to the current abstraction system with respect to the ordering $\sqsupset$.

Lemma 4.9 *For all $j \geq 0$, $M_j \sqsupset M_{j+1}$ whenever S_{j+1} is generated from S_j by an application of* **Coll1**, **Coll2**, **Simpl**, **Ident**, **Shar1**, *or* **Shar2**.

Proof. We consider only the application of **Coll1**, **Ident**, **Shar1**, and **Shar2**. The proof for **Coll2** is very similar to that for **Coll1**, and the proof for **Simpl** is trivial.

Coll1. We can think of S_{j+1} as being derived from S_j by applying the intermediate steps below.

$$\begin{array}{lclclclcl}
S_j & = & T & \cup & \{u \not\equiv v\} & \cup & \{v_1 \equiv s_1[v_2]\} & \cup & \{v_2 \equiv s_2\} \\
S & = & T[v_1/s_2] & \cup & \{u \not\equiv v\}[v_1/v_2] & \cup & \{v_1 \equiv s_1[v_2]\} & \cup & \{v_2 \equiv s_2\} \\
S_{j+1} & = & T[v_1/s_2] & \cup & \{u \not\equiv v\}[v_1/v_2] & \cup & & & \{v_2 \equiv s_2\}
\end{array}$$

As in the proof of Lemma 4.8 we can easily show that S is an abstraction systems as well. Then, where M is the multisets associated to S, we show that $M_j \sqsupseteq M \sqsupset M_{j+1}$.

($M_j \sqsupseteq M$) If v_1 does not occur in T then $M_j = M$, as S_j and S coincide. If v_1 occurs in T, since S_j is an abstraction system, it will necessarily occur in the right-hand side of some equations of T. Let $v_0 \equiv s_0$ be any such equation. Since

$$(v_0 \equiv s_0[v_1]) \prec (v_1 \equiv s_1[v_2]) \prec (v_2 \equiv s_2) \tag{2}$$

we know from Lemma 4.5(1) that every $v \equiv t$ in S such that $(v_2 \equiv s_2) \prec (v \equiv t)$ has a higher height in S_j than $v_0 \equiv s_0$. The replacement of v_1 by s_2 adds an edge from $v_0 \equiv s_0$ only to nodes $v \equiv t$ like the one above. This means that, going from S_j to S, the only new edges are from a node of S_j to one that is already higher. By Lemma 4.5(2) then no node in S_j moves to a greater height in S because of such edge additions. Now, $v_0 \equiv s_0[v_1]$ above becomes $v_0 \equiv s_0[v_1/s_2]$ in S, hence it may become reducible even if it was irreducible before. If n is the height of $v_0 \equiv s_0$ in S, then a pair of the form $(n, 0)$ may be replaced by the larger pair $(n, 1)$ when going from M_j to M. This, however, is not a problem because at least one greater pair, $(n + 1, \mathsf{r}(v_1 \equiv s_1))$, is replaced by a smaller one as well. To see this observe that, since v_1 does not occur in $S \setminus \{v_1 \equiv s_1\}$, the height of $v_1 \equiv s_1$ in S is 0. However, because of $(v_0 \equiv s_0) \prec (v_1 \equiv s_1)$ it was greater than 0 in S_j. By definition of $\sqsupset$, we can conclude that $S_j \sqsupseteq M$.

($M \sqsupset M_{j+1}$) As S_{j+1} is obtained from S by removing the node $v_1 \equiv s_1$, we can use Lemma 4.5(4) to conclude that the pairs corresponding to the remaining nodes do not increase. Since one pair (the one corresponding to $v_1 \equiv s_1$) is removed, we have that $M \sqsupset M_{j+1}$.

Ident. We have that $S_j = T \cup \{x \equiv s, y \equiv t\}$ and $S_{j+1} = T[x/y] \cup \{y \equiv t\}$, where $\mathsf{h}(x \equiv s) \leq \mathsf{h}(y \equiv t)$ in S_j.

The graph induced by S_{j+1} can be obtained from the one induced by S_j as follows. First, add edges from the immediate predecessors in S_j of $x \equiv s$ to $y \equiv t$. Since the height of $y \equiv t$ is at least the height of $x \equiv s$, and thus larger than the height of these predecessors, Lemma 4.5(2) shows that this does not change the height of any node. Second, remove the edges that go from the immediate predecessors in S_j of $x \equiv s$ to $x \equiv s$. By Lemma 4.5(3), this does not increase the height of any node. Third, remove the node $x \equiv s$. By Lemma 4.5(4), this does not increase the height of any of the remaining nodes.

By applying the substitution $[x/y]$ to the equations in T, the reducibility of a node containing x may change from 0 to 1. However, these nodes have a height that is smaller than the height of $x \equiv s$. Thus, an increase in the pair associated to such a node in the multiset is compensated by the fact that the pair associated to $x \equiv s$ is removed. This shows that $M_j \sqsupset M_{j+1}$.

Shar1. We know that S_j and S_{j+1} have the following form:

$$\begin{array}{lclcccccccc} S_j & = & T & & & \cup & \{u \not\equiv v\} & \cup & \{x \equiv t\} & \cup & \{\bar{y}_1 \equiv \bar{r}_1\} \\ S_{j+1} & = & T[x/s(\bar{y},\bar{z})[\bar{y}_1/\bar{r}_1]] & \cup & \{\bar{z} \equiv \bar{r}\} & \cup & \{u \not\equiv v\} & \cup & \{x \equiv s(\bar{y},\bar{r})\} & \cup & \{\bar{y}_1 \equiv \bar{r}_1\} \end{array}$$

Observe that there may be more nodes in S_{j+1} than in S_j: those corresponding to the equations in $\bar{z} \equiv \bar{r}$. Let n be the height of $x \equiv t$ in S_j, which is at least 1 as x occurs in T by assumption. We start by showing that the height of the new nodes in S_{j+1} cannot be greater than n.

Going from S_j to S_{j+1}, the new equations $\bar{z} \equiv \bar{r}$ are introduced while each occurrence of x in the right-hand side of an equation is replaced by $s(\bar{y},\bar{z})[\bar{y}_1/\bar{r}_1]$. Consider any equation $z \equiv r$ in $\bar{z} \equiv \bar{r}$. Observing that z occurs in the tuple $\bar{z}$, we then obtain

$$\varphi[x/s(\bar{y},\bar{z})[\bar{y}_1/\bar{r}_1]] \quad \prec_{j+1} \quad (z \equiv r)$$

for all equations φ (and only those) such that

$$\varphi \quad \prec_j \quad (x \equiv t).$$

Using the fact that $\prec_j$ is acyclic, it is easy to see that no such equation φ changes its height when going from S_j to S_{j+1}. As a consequence, $z \equiv r$ has in S_{j+1} the height that $x \equiv t$ had in S_j, namely, n.

The new node $z \equiv r$ may also have outgoing edges. Since the variables in $\mathcal{V}ar(s(\bar{y}, \bar{r})) \setminus \mathcal{V}ar(t)$ do not occur in S_j, however, these edges will go only into old nodes ψ such that $x \equiv t \prec_j \psi$. In other words, all the edges out of $z \equiv r$ will end in nodes whose height was already $> n$ in S_j.

Similarly, the replacement of x by $s(\bar{y}, \bar{z})[\bar{y}_1/\bar{r}_1]$ in T may introduce new edges in S_{j+1} between old nodes,[21] but it is again easy to see that each of these edges will go from a node to one with already greater height. Finally, and again because the variables in $\mathcal{V}ar(s(\bar{y}, \bar{r})) \setminus \mathcal{V}ar(t)$ do not occur in S_j, the replacement of t by $s(\bar{y}, \bar{r})$ in the node $x \equiv t$ will possibly remove some edges from S_{j+1}, but will not introduce new ones.

By Points 1 and 3 of Lemma 4.5 then some old nodes may move to a lower height in S_{j+1} but none will move to a higher height because of the mentioned replacements. In conclusion, we can say that the number of nodes at heights $> n$ will not increase from S_j to S_{j+1}. In addition, the reducibility value of these nodes will not change (since their right-hand sides are not modified).

Now, if some node with height $> n$ in S_j moves to a smaller height in S_{j+1}, we can already conclude that $M_j \sqsupset M_{j+1}$. If, on the other hand, all the nodes at height $> n$ keep the same height, to prove that $M_j \sqsupset M_{j+1}$ we argue that the number of reducible nodes at height n decreases. To see that it is enough to make the following three observations. First, it is possible that the replacement of x by $s(\bar{y}, \bar{z})$ alters the reducibility of some nodes to 1, but as shown above this will happen only at heights $< n$. Second, when no old node at height $> n$ moves to a smaller height, the number of nodes at height n increases only because of the presence of the new nodes in $\bar{z} \equiv \bar{r}$, whose reducibility is 0, as each $r \in \bar{r}$ is in G_i. Third, the node $x \equiv t$ of S_j, which by assumption had height $n > 0$ and was reducible, is replaced by the node $x \equiv s(\bar{y}, \bar{r})$ whose height in S_{j+1} is 0.

Shar2. We know that S_j and S_{j+1} have the following form:

$$\begin{array}{lclcccccc} S_j & = & T & \cup & \{u \not\equiv v\} & \cup & \{x \equiv t[\bar{y}_1]\} & \cup & \{\bar{y}_1 \equiv \bar{r}_1\} \\ S_{j+1} & = & T[x/s[\bar{y}_1/\bar{r}_1]] & \cup & \{u \not\equiv v\} & \cup & \{x \equiv s[\bar{y}_1/\bar{r}_1]\} & \cup & \{\bar{y}_1 \equiv \bar{r}_1\} \end{array}$$

Let n be the height of $x \equiv t$ in S_j. As in the **Shar1** case we can show that the number of nodes at height $> n$ does not increase going from S_j to S_{j+1}, and the reducibility value of these nodes does not change. It is enough to show then that the number of reducible nodes at height n decreases by one. But this is an immediate consequence of the fact that the node $x \equiv t$ in S_j, which by assumption had height $n > 1$ and was reducible, is replaced by the node $x \equiv s(\bar{y}, \bar{r})$ whose height in S_{j+1} is 0. □

Proposition 4.10 (Termination) *The combination procedure halts on all inputs.*

[21] Specifically, between a node of the form $x_0 \equiv t_0[x]$ and a successor node of one of the equations in $\bar{y}_1 \equiv \bar{r}_1$.

Proof. First we need to make sure that every step of the procedure is executable in finite time. This is immediate for Step 3 and true for Step 1 because the purification procedure used to produce $AS(s_0 \equiv t_0)$ always terminates.

It is true for each iteration of Step 2 if the preconditions of the derivation rules can be tested in finite time. Given that the word problem in E_1 and in E_2 is decidable, this is immediate for all the rules in Fig. 2. For **Shar1**, it should be clear that the test on the preconditions (a), (e) and (f) is effective. The test on conditions (b) and (d) is effective because G_i is recursive by assumption. The computation of the normal form of t in (c) is effective because G_i-normal forms are computable for $i = 1, 2$ by assumption; its decompositions into the terms $s, \bar{r}$ is effective by Proposition 3.12 because G_i is recursive. A similar argument applies to the preconditions of **Shar2**.

To conclude the proof all we need to show is that the procedure applies the various rules only finitely many times. But this is an immediate consequence of Lemma 4.9 and the well-foundedness of $\sqsupset$. $\square$

The next two lemmas show that the derivation rules preserve satisfiability.

Lemma 4.11 *Let $\bar{v}_{j-1}$ be a sequence consisting of the left-hand side variables of S_{j-1} and $\bar{v}_j$ be a sequence consisting of the left-hand side variables of S_j. Then, $\exists \bar{v}_{j-1}.S_{j-1} \leftrightarrow \exists \bar{v}_j.S_j$ is valid in E.*

Proof. We can index all the possible cases by the derivation rule applied to S_{j-1} to obtain S_j.[22] Let $\mathcal{A}$ be any model of E.

First assume that S_j has been produced by an application of **Coll1**. We know that S_{j-1} and S_j have the form

$$\begin{array}{lllllll} S_{j-1} & = & \{u \not\equiv v\} & \cup & \{x \equiv t[y]\} & \cup & \{y \equiv r\} \;\cup\; T \\ S_j & = & \{u \not\equiv v\}[x/y] & \cup & & & \{y \equiv r\} \;\cup\; T[x/r] \end{array}$$

and that $y =_{E_i} t$ for $i = 1$ or $i = 2$.

Let α be a valuation of V satisfying S_{j-1} in $\mathcal{A}$. It is enough to show that there exists a valuation α' that satisfies S_j in $\mathcal{A}$ and coincides with α on the free variables of $\exists \bar{v}_{j-1}.S_{j-1} \leftrightarrow \exists \bar{v}_j.S_j$.

Since $y \equiv t$ is valid in E, for being valid in E_i, α must assign both x and y with $[\![t]\!]^{\mathcal{A}}_{\alpha}$, i.e., the interpretation of the term t in $\mathcal{A}$ under the valuation α. In addition, since α satisfies S_{j-1}, we know that $\alpha(y) = [\![r]\!]^{\mathcal{A}}_{\alpha}$. It follows immediately that α satisfies S_j in $\mathcal{A}$. Thus, we can take $\alpha' := \alpha$.

Now, assume that the valuation α satisfies S_j in the model $\mathcal{A}$ of E. Again, we must show that there exists a valuation α' that satisfies S_{j-1} in $\mathcal{A}$ and coincides with α on the free variables of $\exists \bar{v}_{j-1}.S_{j-1} \leftrightarrow \exists \bar{v}_j.S_j$.

[22]Ignoring the trivial case in which S_j coincides with S_{j-1}.

Observe that, since S_{j-1} is an abstraction system, x does not occur in $y \equiv r$, and as a consequence it does not occur in S_j at all. Let α' be the valuation defined by $\alpha'(z) := \alpha(z)$ for all $z \neq x$ and $\alpha'(x) := \alpha(y)$. It is immediate that α' satisfies the set $T_1 := T \cup \{x \equiv r\} \cup \{u \not\equiv v\} \cup \{x \equiv y\} \cup \{y \equiv r\}$ in $\mathcal{A}$. Since $\mathcal{A}$ is a model of E and the equation $y \equiv t$ is valid in E, it is also immediate that α' satisfies the set $T_2 := \{x \equiv t\}$ in $\mathcal{A}$. It follows that α' satisfies S_{j-1}, which is a subset of $T_1 \cup T_2$. Since α and α' differ only w.r.t. the value they assign to x, and x is a left-hand side variable in S_{j-1} and does not occur in S_j, this completes the proof that $\exists \bar{v}_{j-1}.S_{j-1} \leftrightarrow \exists \bar{v}_j.S_j$ is valid in E.

The proof for **Coll2** can be derived as a special case of the one for **Coll1** with r replaced by y. **Ident** can be treated similarly.

When S_j is generated by an application of **Simpl**, S_{j-1} and S_j have the form

$$\begin{array}{lcl} S_{j-1} & = & T \ \cup \ \{x \equiv t\} \\ S_j & = & T \end{array}$$

with $x \notin \mathcal{V}ar(T)$. It immediate that if S_{j-1} is satisfied by a valuation α in $\mathcal{A}$, so is S_j. Conversely, assume that S_j is satisfied in $\mathcal{A}$ by some valuation α. Let α' be a valuation coinciding with α on all variables except x. For the variable x, let $\alpha'(x) := [\![t]\!]^{\mathcal{A}}_{\alpha}$. From the assumptions and the fact that S_{j-1} is an abstraction system, we know that x is not in $\mathcal{V}ar(t) \cup \mathcal{V}ar(T)$. This, together with the definition of $\alpha'(x)$, implies that α' satisfies S_{j-1}. In addition, α and α' coincide on the free variables of $\exists \bar{v}_{j-1}.S_{j-1} \leftrightarrow \exists \bar{v}_j.S_j$ since x is a left-hand side variable in S_{j-1} and does not occur in S_j.

When S_j is generated by an application of **Shar1**, S_{j-1} and S_j have the form

$$\begin{array}{lclclclclcl} S_{j-1} & = & T & & & \cup & \{u \not\equiv v\} & \cup & \{x \equiv t\} & \cup & \{\bar{y}_1 \equiv \bar{r}_1\} \\ S_j & = & T[x/s(\bar{y},\bar{z})[\bar{y}_1/\bar{r}_1]] & \cup & \{\bar{z} \equiv \bar{r}\} & \cup & \{u \not\equiv v\} & \cup & \{x \equiv s(\bar{y},\bar{r})\} & \cup & \{\bar{y}_1 \equiv \bar{r}_1\} \end{array}$$

Let $\mathcal{A}$ be any model of E. First, assume that some valuation α of V satisfies S_j in $\mathcal{A}$. Since S_j contains the equation $x \equiv s(\bar{y},\bar{r})$ and $t =_E s(\bar{y},\bar{r})$, we know that $\alpha(x) = [\![t]\!]^{\mathcal{A}}_{\alpha}$. In addition, since S_j also contains the equations $\bar{y}_1 \equiv \bar{r}_1$ and $\bar{z} \equiv \bar{r}$, we also know that $\alpha(x) = [\![s(\bar{y},\bar{z})[\bar{y}_1/\bar{r}_1]]\!]^{\mathcal{A}}_{\alpha}$. Obviously, this implies that α satisfies S_{j-1} in $\mathcal{A}$.

Conversely, assume that some valuation α satisfies S_{j-1} in $\mathcal{A}$. Let α' be a valuation coinciding with α on all variables except those in $\bar{z}$. For each component $z_i \equiv r_i$ of $\bar{z} \equiv \bar{r}$ we define $\alpha'(z_i) := [\![r_i]\!]^{\mathcal{A}}_{\alpha}$. As above, it is easy to show that $\alpha'(x) = \alpha(x) = [\![s(\bar{y},\bar{r})]\!]^{\mathcal{A}}_{\alpha'}$ and $\alpha'(x) = [\![s(\bar{y},\bar{z})[\bar{y}_1/\bar{r}_1]]\!]^{\mathcal{A}}_{\alpha'}$. This implies that α' satisfies S_j in $\mathcal{A}$. Since the variables in $\bar{z}$ are left-hand side variables of S_j, which do not occur in S_{j-1}, the valuations α and α' coincide on the free variables of $\exists \bar{v}_{j-1}.S_{j-1} \leftrightarrow \exists \bar{v}_j.S_j$.

The proof for **Shar2** is almost identical to the one for **Shar1**. $\square$

The lemma above immediately entails the following weaker lemma (see the comment following Proposition 4.3).

Lemma 4.12 *For all $j > 0$, the abstraction system S_j is satisfiable in E iff S_{j-1} is satisfiable in E.*

It is now easy to show that the combination procedure is sound.

Proposition 4.13 (Soundness) *If the combination procedure succeeds on an input (s_0, t_0), then $s_0 =_E t_0$.*

Proof. Let $\mathcal{S}$ be the sequence of abstraction systems generated by the procedure on input (s_0, t_0). By the procedure's definition, we know that, if the procedure succeeds, there is an $n > 0$ such that $S_n = \{v \not\equiv v\} \cup T$. Since S_n is clearly unsatisfiable in E, we can conclude by a repeated application of Lemma 4.12 that $S_0 = AS(s_0 \not\equiv t_0)$ is also unsatisfiable in E. By Proposition 4.3, it follows that $s_0 \not\equiv t_0$ is unsatisfiable in E, which means that $s_0 =_E t_0$. □

The completeness proof will be simplified by appealing to the following lemma.

Lemma 4.14 *The final abstraction system S generated by a failed execution of the combination procedure can be partitioned into the sets*

$$D := \{x \not\equiv y\} \qquad T_1 := \{u_j \equiv r_j\}_{j \in J} \qquad T_2 := \{v_k \equiv t_k\}_{k \in K}$$

where

1. *x and y are distinct, and J and K are finite;*
2. *each $r_j \in T(\Sigma_1, V) \setminus V$ and each $t_k \in T(\Sigma_2, V) \setminus V$;*
3. *each u_j occurs only once in T_1 and each v_k occurs only once in T_2;*
4. *for all $v \in \mathcal{V}ar(T_1) \cap \mathcal{V}ar(T_2)$,*
   (a) *if $v = u_j$ for some $j \in J$ then $v \in \mathcal{V}ar(t_k)$ for some $k \in K$,*
   if $v = v_k$ for some $k \in K$ then $v \in \mathcal{V}ar(r_j)$ for some $j \in J$,
   (b) *if $v = u_j$ for some $j \in J$ then $r_j \in G_1$,*
   if $v = v_k$ for some $k \in K$ then $t_k \in G_2$.

Proof. Since the procedure has failed, we know that $x \neq y$, and thus point 1 is trivial. Points 2, 3, 4a are an immediate consequence of the fact that S is an abstraction system.

To prove (4b), let $v = u_j$ for some $j \in J$ and notice that v occurs in $S \setminus \{x \not\equiv y, u_j \equiv r_j\}$. Assume by contradiction that r_j is not an element of G_1. Then we can also assume with no loss of generality, since $\prec$ is acyclic and S is finite, that there are no equations $v_k \equiv t_k$ in S such that $(u_j \equiv r_j) \prec (v_k \equiv t_k)$ and $t_k \notin G_2$.[23] But then, it is not difficult to see that one of **Coll1**, **Coll2**, **Shar1**, **Shar2** applies to $u_j \equiv r_j$, against the assumption that S is the final abstraction system. If $v = v_k$ the argument is analogous. □

To prove that the procedure is complete for the word problem in $E := E_1 \cup E_2$ we make the additional assumption that

$$E_1{}^{\Sigma} = E_2{}^{\Sigma}.$$

In this case, we have the following.

Proposition 4.15 (Completeness) *The combination procedure succeeds on input (s_0, t_0) if $s_0 =_E t_0$.*

Proof. By Lemma 4.10, the procedure either succeeds or fails; therefore, we can prove the claim by proving that whenever the procedure fails on input (s_0, t_0), the formula $s_0 \not\equiv t_0$ is satisfiable in E. Thus, assume that the procedure fails and let S_n be the abstraction system generated by the last rule application. Given Lemma 4.12 and the construction of S_0, it is enough to show that S_n is satisfiable in E.

From Lemma 4.14 we know that S_n is an abstraction system with an initial formula of the form $x \not\equiv y$, where x and y are distinct. Furthermore, $S_n \setminus \{x \not\equiv y\}$ can be partitioned into the sets

$$T_1 \;:=\; \{u_j \equiv r_j\}_{j \in J} \quad \text{and} \quad T_2 \;:=\; \{v_k \equiv t_k\}_{k \in K},$$

where T_1 and T_2 satisfy Lemma 4.14(1–4b). For $i = 1, 2$, let $\mathcal{A}_i$ be a Σ_i-algebra free in E_i over a countably infinite set X_i and α_i a bijective valuation of V onto X_i. From Corollary 3.8 we know that $\mathcal{A}_i{}^{\Sigma}$ is free in $E_i{}^{\Sigma}$ over $Y_i := [\![G_i]\!]^{\mathcal{A}_i}_{\alpha_i}$ and $X_i \subseteq Y_i$.

Now, for $i = 1, 2$, we will construct a valuation β_i of $\mathcal{V}ar(T_i)$ into $\mathcal{A}_i$ that assigns with a distinct element of Y_i each variable shared by $\{x \not\equiv y\} \cup T_1$ and T_2. Furthermore, β_1 and β_2 will be such that

$$\mathcal{A}_1, \beta_1 \models \{x \not\equiv y\} \cup T_1 \quad \text{and} \quad \mathcal{A}_2, \beta_2 \models T_2.$$

[23] Otherwise, we can consider first the case in which $v = v_k$ since v_k is also a shared variable.

By Proposition 3.17 then, this will entail that $\{x \not\equiv y\} \cup T_1 \cup T_2$ (that is, S_n) is satisfiable in E. We can restrict our attention to the case in which $i = 1$, as the other case (which is even simpler) can be treated analogously.

Let β_1 be the valuation of $\mathcal{V}ar(T_1)$ defined as follows:

$$\beta_1(v) := \begin{cases} \alpha_1(v) & \text{for all } v \in \bigcup_j \mathcal{V}ar(r_j) \\ [\![r_j]\!]^{\mathcal{A}_1}_{\alpha_1} & \text{for all } v \in \bigcup_j \{u_j\} \end{cases}$$

Such a valuation is well-defined because all the variables u_j are distinct and none of them belongs to $V_1 := \bigcup_j \mathcal{V}ar(r_j)$, as shown in Lemma 4.14. By construction, β_1 satisfies T_1 in $\mathcal{A}_1$. We prove below that β_1 is injective.

Let $u, v \in \mathcal{V}ar(T_1)$, $u \neq v$. If both u and v are in V_1, then $\beta_1(u) \neq \beta_1(v)$ by construction of α_1. Hence, let $u = u_j$ for some $j \in J$ and assume by contradiction that $\beta_1(u_j) = \beta_1(v)$.

If $v = u_\ell$ for some $\ell \in J$, then $\mathcal{A}_1, \beta_1 \models r_j \equiv r_\ell$ by construction of β_1. As β_1 evaluates the variables in the equation $r_i \equiv r_j$ by distinct generators of $\mathcal{A}_1$, and $\mathcal{A}_1$ is E_1-free, we obtain that $r_j =_{E_1} r_\ell$ by Proposition 2.1; but then, since either $\mathsf{h}(u_\ell \equiv r_\ell) \leq \mathsf{h}(u_j \equiv r_j)$ or $\mathsf{h}(u_j \equiv r_j) \leq \mathsf{h}(u_\ell \equiv r_\ell)$, **Ident** applies to S_n against the assumption that S_n is the final abstraction system.

If $v \in V_1$, similarly to the previous case, we can show that $v =_{E_1} r_j$ and (since E_1 is non-trivial) that v occurs in r_j. Therefore, either **Coll1** or **Coll2** applies, again against the assumption that S_n is the final abstraction system. In conclusion, β_1 is injective.

We now show that $\beta_1(v) \in Y_1$ for every variable v that T_1 shares with T_2. Let $v \in \mathcal{V}ar(T_1) \cap \mathcal{V}ar(T_2)$. If $v \in V_1$, then $\beta_1(v) = \alpha_1(v) \in X_1 \subseteq Y_1$ by construction. If $v = u_j$ for some $j \in J$, we know from Lemma 4.14(4b) that $r_j \in G_1$. Observing that β_1 assigns the variables of r_j with elements of X_1 and recalling the definition of Y_1, we can conclude that $\beta_1(v)$, which is the same as $[\![r_j]\!]^{\mathcal{A}_1}_{\alpha_1}$, is an element of Y_1.

To complete the proof we finally need to make sure that β_1 is properly defined for x and y as well. If both x and y occur in T_1, we know by the above that β_1 is already defined for them and that $\beta_1(x) \neq \beta_1(y)$, as x and y are distinct. If x occurs in T_2 as well, we also know that $\beta_1(x) \in Y_1$ (similarly for y). If x or y (or both) does not occur in T_1, let $Z := \{x, y\} \setminus \mathcal{V}ar(T_1)$. Since Y_1 is infinite, we can extend β_1 arbitrarily to $\mathcal{V}ar(T_1) \cup Z$ so that, for all $z \in Z$, $\beta_1(z) \in Y_1$ and $\beta_1(z) \neq \beta_1(v)$ for all $v \in \mathcal{V}ar(T_1) \cup Z \setminus \{z\}$.

In conclusion, we have constructed a valuation β_1 of $\mathcal{V}ar(T_1) \cup \{x, y\}$ that satisfies $\{x \not\equiv y\} \cup T_1$ in $\mathcal{A}_1$ and maps the variables shared by $\{x \equiv y\} \cup T_1$ and T_2 to distinct elements of Y_1. $\square$

As in [BT99], nowhere in the proof of Proposition 4.15 did we use the fact that the rule **Simpl** could no longer be applied. This means that the combination procedure obtained by removing **Simpl** is also complete. Obviously, this modified procedure is still sound and terminating.

Combining the results of this section, we obtain the following modularity result for the decidability of the word problem.

Theorem 4.16 *Let E_1, E_2 be two non-trivial equational theories of signature Σ_1, Σ_2, respectively, such that $\Sigma := \Sigma_1 \cap \Sigma_2$ is a set of constructors for both E_1 and E_2, and ${E_1}^{\Sigma} = {E_2}^{\Sigma}$. Let G_1, G_2 be Σ-bases of E_1, E_2, respectively. If for $i = 1, 2$,*

- *G_i is closed under bijective renaming of V and recursive,*
- *G_i-normal forms are computable for Σ and E_i, and*
- *the word problem in E_i is decidable,*

then the word problem in $E_1 \cup E_2$ is also decidable.

We would like to point out that the corresponding result in [BT99] is indeed a corollary of the above. 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. In [BT99], 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, as shown in the same paper, that is always the case whenever $G_{E_i}(\Sigma, V)$-normal forms are computable for Σ and E_i.

Similarly to [BT99], the decidability result of Theorem 4.16 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, as we show in the next section.

5 Modularity of Constructors

In this section, we show that the property of being a set of constructors is preserved by the union of theories. We also show that normal forms are computable in a union theory whenever they are computable in its component theories and the word problem is decidable for those theories.

For this purpose, let us fix two non-trivial equational theories E_1, E_2 with respective signatures Σ_1, Σ_2 such that, for $i = 1, 2$

- $\Sigma := \Sigma_1 \cap \Sigma_2$ is a set of constructors for E_i;
- ${E_1}^{\Sigma} = {E_2}^{\Sigma}$;

- E_i admits a Σ-base G_i closed under bijective renaming of V;
- G_i is recursive and G_i-normal forms are computable for Σ and E_i.
- the word problem for E_i is decidable.

To simplify the proofs we will also assume, without loss of generality, that if $s(\bar{r}) \in T(\Sigma, G_i)$ is the normal form $NF^{\Sigma}_{E_i}(t)$ of a Σ_i-term t $(i = 1, 2)$, no non-variable term in $\bar{r}$ is equivalent in E_i to a variable. To see that there is no loss of generality first notice that the only variables a Σ_i-term that can collapse to in E_i are its own variables—otherwise E_i would be trivial. Since the word problem for E_i is decidable, it is decidable whether a term is equivalent in E_i to one of its variables. But then, we can effectively build a function that given a Σ_i-term t, first computes $s(\bar{r}) = NF^{\Sigma}_{E_i}(t)$ and then replaces each collapsing term in $\bar{r}$ by the variable it collapses to. If $\bar{r}'$ is the tuple obtained after all such replacements, it is clear that $s(\bar{r}')$ is also a G-normal form of t.

Now, let

$$E \ := \ E_1 \cup E_2.$$

We will show below that $E^{\Sigma} = {E_1}^{\Sigma} = {E_2}^{\Sigma}$ and Σ is a set of constructors for E. Moreover, E admits a Σ-base G such that G is recursive and closed under bijective renamings, and G-normal forms are computable for Σ and E.

We will use a particular model of E, obtained as a fusion of the free models of E_1 and E_2. More precisely, for $i = 1, 2$, let us fix a Σ_i-algebra $\mathcal{A}_i$ free in E_i over a countably infinite set X_i. Let us also fix an arbitrary bijective valuation α_i of V onto X_i, and consider the set

$$Y_i \ := \ [\![G_i]\!]^{\mathcal{A}_i}_{\alpha_i}.$$

We know from Corollary 3.8 that $X_i \subseteq Y_i$ and ${\mathcal{A}_i}^{\Sigma}$ is free in ${E_i}^{\Sigma}$ over Y_i. Observe that $\mathcal{A}_i$ is countably infinite, given our assumption that X_i is countably infinite and Σ_i is countable. As a consequence, Y_i is countably infinite as well.

Now let $Z_{i,2} := Y_i \setminus X_i$ for $i = 1, 2$, and let $\{Z_{1,1}, Z_1\}$ be a partition of X_1 such that Z_1 is countably infinite and $Card(Z_{1,1}) = Card(Z_{2,2})$.[24] Similarly, let $\{Z_{2,1}, Z_2\}$ be a partition of X_2 such that $Card(Z_{2,1}) = Card(Z_{1,2})$ and Z_2 is countably infinite. Then consider 3 arbitrary bijections

$$h_1 \colon Z_{1,2} \longrightarrow Z_{2,1}, \quad h_2 \colon Z_1 \longrightarrow Z_2, \quad h_3 \colon Z_{1,1} \longrightarrow Z_{2,2},$$

as shown in Figure 4. Observing that $\{Z_{i,1}, Z_i, Z_{i,2}\}$ is a partition of Y_i for each i, it is immediate that $h_1 \cup h_2 \cup h_3$ is a well-defined bijection of Y_1 onto Y_2.

[24] This is possible because $Z_{2,2}$ is countable (possibly finite).

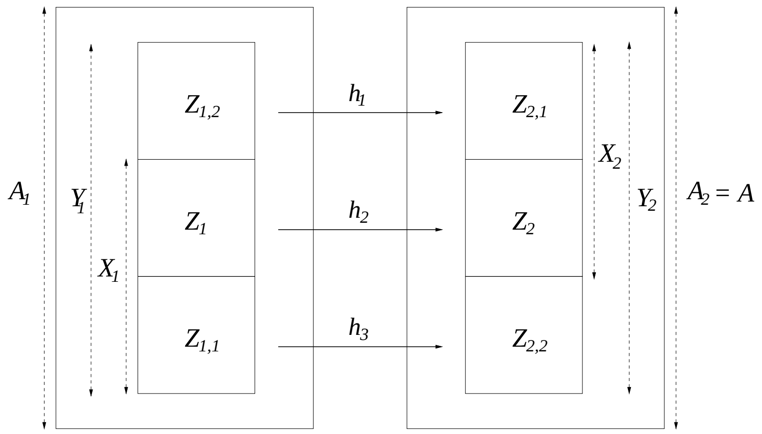

Figure 4: The Fusion $\mathcal{A}$ of $\mathcal{A}_1$ and $\mathcal{A}_2$.

Since ${E_1}^{\Sigma} = {E_2}^{\Sigma}$ and both Y_1 and Y_2 are countably infinite, ${\mathcal{A}_1}^{\Sigma}$ and ${\mathcal{A}_2}^{\Sigma}$ are both free in the same Σ-variety over sets with the same cardinality. It follows by Proposition 2.2 then, that the bijection $h_1 \cup h_2 \cup h_3$ can be extended to a Σ-isomorphism h of ${\mathcal{A}_1}^{\Sigma}$ onto ${\mathcal{A}_2}^{\Sigma}$.

Lemma 5.1 *The Σ-isomorphism h induces a fusion $\mathcal{A}$ of $\mathcal{A}_1$ and $\mathcal{A}_2$ such that the following holds:*

1. *$\mathcal{A}^{\Sigma_1}$ is free in E_1 over $X_1' := Z_{2,2} \cup Z_2$;*
2. *$\mathcal{A}^{\Sigma_2}$ is free in E_2 over $X_2 = Z_{2,1} \cup Z_2$;*
3. *$\mathcal{A}^{\Sigma}$ is free in ${E_1}^{\Sigma} = {E_2}^{\Sigma}$ over $Y_2 = Z_{2,1} \cup Z_2 \cup Z_{2,2}$.*
4. *$Y_2 = [\![G_2]\!]^{\mathcal{A}^{\Sigma_2}}_{\alpha_2} = [\![G_1]\!]^{\mathcal{A}^{\Sigma_1}}_{\beta_1}$, with $\beta_1 := h \circ \alpha_1$.*

Proof. By Proposition 3.2, there is a fusion $\mathcal{A}$ of $\mathcal{A}_1$ and $\mathcal{A}_2$ such that the identity on the carrier of $\mathcal{A}_2$ is a Σ_2-isomorphism of $\mathcal{A}_2$ onto $\mathcal{A}^{\Sigma_2}$, and h is a Σ_1-isomorphism of $\mathcal{A}_1$ onto $\mathcal{A}^{\Sigma_1}$. The first three points then are an immediate consequence of the construction of h and the choice of $\mathcal{A}$. By definition, $Y_2 = [\![G_2]\!]^{\mathcal{A}^{\Sigma_2}}_{\alpha_2}$. Let $Y_1' := [\![G_1]\!]^{\mathcal{A}^{\Sigma_1}}_{\beta_1}$. We prove the last point by showing that Y_1' is included in Y_2 and vice versa.

($Y_1' \subseteq Y_2$) Let y be an element of Y_1'. By definition of Y_1' then, there is a term r in G_1 such that $y = [\![r]\!]^{\mathcal{A}^{\Sigma_1}}_{\beta_1}$. It r is a variable v, then $y = [\![r]\!]^{\mathcal{A}^{\Sigma_1}}_{\beta_1} = \beta_1(v) = h(\alpha_1(v))$. Since $\alpha_1(v)$ is in X_1 by definition of α_1, we have that $h(\alpha_1(v))$ is in $X_1' \subseteq Y_2$ by construction of h.

It r is a not variable, let $v_1, \ldots, v_n$ be r's variables. Then,

$$
\begin{array}{rcll}
y = [\![r]\!]^{\mathcal{A}^{\Sigma_1}}_{\beta_1} & = & r^{\mathcal{A}^{\Sigma_1}}(\beta_1(v_1), \ldots, \beta_1(v_n)) & \text{(by def. of } [\![]\!]) \\
& = & r^{\mathcal{A}^{\Sigma_1}}(h(\alpha_1(v_1)), \ldots, h(\alpha_1(v_n))) & \text{(by def. of } \beta_1) \\
& = & h(r^{\mathcal{A}_1}(\alpha_1(v_1), \ldots, \alpha_1(v_n))) & (h\ \Sigma_1\text{-isomorph.}) \\
& = & h([\![r]\!]^{\mathcal{A}_1}_{\alpha_1}) & \text{(by def. of } [\![]\!])
\end{array}
$$

Since $r \in G_1$, the element $[\![r]\!]^{\mathcal{A}_1}_{\alpha_1}$ is in Y_1 by definition of Y_1. It follows by construction of h that $y = h([\![r]\!]^{\mathcal{A}_1}_{\alpha_1})$ is in Y_2.

($Y_2 \subseteq Y_1'$) Let y be an element of Y_2. By construction of h, $h^{-1}(y)$ is an element of Y_1, which means that there is a term r in G_1 such that $h^{-1}(y) = [\![r]\!]^{\mathcal{A}_1}_{\alpha_1}$. If r is not a variable, let $v_1, \ldots, v_n$ be the variables of r. Then,

$$
\begin{array}{rcll}
y = h(h^{-1}(y)) & = & h([\![r]\!]^{\mathcal{A}_1}_{\alpha_1}) & \\
& = & h(r^{\mathcal{A}_1}(\alpha_1(v_1), \ldots, \alpha_1(v_n))) & \text{(by def. of } [\![]\!]) \\
& = & r^{\mathcal{A}^{\Sigma_1}}(h(\alpha_1(v_1)), \ldots, h(\alpha_1(v_n))) & (h\ \Sigma_1\text{-isomorph.}) \\
& = & r^{\mathcal{A}^{\Sigma_1}}(\beta_1(v_1), \ldots, \beta_1(v_n)) & \text{(by def. of } \beta_1) \\
& = & [\![r]\!]^{\mathcal{A}^{\Sigma_1}}_{\beta_1} & \text{(by def. of } [\![]\!])
\end{array}
$$

If r is a variable, then $y = h(h^{-1}(y)) = h([\![r]\!]^{\mathcal{A}_1}_{\alpha_1}) = h(\alpha_1(r)) = \beta_1(r) = [\![r]\!]^{\mathcal{A}^{\Sigma_1}}_{\beta_1}$. In either case, it follows by definition of Y_1' that y is in Y_1'. □

The first interesting property of $E := E_1 \cup E_2$ is that it is a conservative extension of both E_1 and E_2.

Proposition 5.2 *For all $j \in \{1, 2\}$ and $t_1, t_2 \in T(\Sigma_j, V)$*

$$t_1 =_{E_j} t_2 \quad \textit{iff} \quad t_1 =_E t_2.$$

Proof. The implication from left to right is immediate since $E_j \subseteq E$. For the converse, assume that $j = 2$ (the proof for $j = 1$ follows by symmetry), and let $t_1, t_2 \in T(\Sigma_2, V)$ such that $t_1 =_E t_2$.

Consider then the fusion $\mathcal{A}$ of $\mathcal{A}_1$ and $\mathcal{A}_2$ constructed above, and recall that $\mathcal{A}^{\Sigma_2}$ is free in E_2 over X_2. Since $t_1 =_E t_2$ and $\mathcal{A}$ is a model of E, we have that $\mathcal{A}, \alpha \models t_1 \equiv t_2$ for any valuation α of $\mathcal{V}ar(t_1 \equiv t_2)$ into A. In particular, we can choose α to be an injection into X_2. Observing that t_1, t_2 are Σ_2-terms we then have that $\mathcal{A}^{\Sigma_2}, \alpha \models t_1 \equiv t_2$. It follows by Proposition 2.1 that $t_1 =_{E_2} t_2$. □

The following is an immediate consequence of the above result.

Corollary 5.3 $E^{\Sigma} = {E_1}^{\Sigma} = {E_2}^{\Sigma}$.

Input: Abstraction system S.

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

Figure 5: A variant of the combination procedure

To show that Σ is a set of constructors for E, we will show that the set of $(\Sigma_1 \cup \Sigma_2)$-terms defined below is a Σ-base of E.

In the following, if r is a $(\Sigma_1 \cup \Sigma_2)$-term we will denote by $\widehat{r}$ the pure term obtained from r by abstracting its alien subterms as done by the purification procedure in Section 2.

Definition 5.4 *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^* such that the following holds:*
   - *(a) $r(\bar{v}) \neq_E v$ for all variables $v \in V$;*
   - *(b) $\widehat{r}_k \notin T(\Sigma_i, V)$ for all non-variable components r_k of $\bar{r}$;*
   - *(c) the tuples $\bar{v}$ and $\bar{r}$ have the same length;*
   - *(d) $r_k \neq_E r_\ell$ 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. The elements of G^* have a stratified recursive structure. A term in $G^* \cap V$ has just one layer. A term $r(\bar{r})$ in $G^* \setminus V$ has at least two layers. Layer 1, the top layer, is made of the term r only; layer 2 is made of all the terms that are at layer 1 in an element of $\bar{r}$; and so on. Furthermore, terms in the same layer all belong to G_1 or to G_2, and if the terms in one layer are in G_i then the terms in the next layer are not in G_i. We will say that a term such as $r(\bar{r})$ above *starts* with the term r.

Also notice that, like each G_i, G^* is closed under variable renaming. We show below that it is recursive as well.

Proposition 5.5 *It is decidable whether a* $(\Sigma_1 \cup \Sigma_2)$*-term is in* G^* *or not.*

Proof. Since $V \subseteq G^*$ by construction, it is enough to consider only non-variable terms. Let $t \in T(\Sigma_1 \cup \Sigma_2, V) \setminus V$. We prove the claim by induction on the rank of t (cf. Definition 2.3).

(Base case) Assume that rank of t is zero, which means that t is a Σ_i-term for $i = 1$ or $i = 2$. From Definition 5.4 then it is easy to see that t is in G^* iff it is in G_i and t is not equivalent in E to a variable. But this is decidable because each G_i is recursive by assumption, E is non-trivial as seen in Subsection 3.3, and the word problem in E is decidable by Theorem 4.16.

(Induction Step) If the rank of t is greater than zero, it is clearly effectively possible to decompose t into terms $r(\bar{v})$ and $\bar{r}$ such that

- $\bar{v}$ and $\bar{r}$ have the same length and each of them has no repeated elements;
- $r(\bar{v}) \in T(\Sigma_i, V) \setminus V$ for $i = 1$ or $i = 2$;
- $\widehat{r}_k \notin T(\Sigma_i, V)$ for all non-variable components r_k of $\bar{r}$;
- $t = r(\bar{r})$.

Notice that although in general such a decomposition is not unique,[25] there are only finitely many of them.[26] Now, for each of these decompositions we can decide whether $r(\bar{v})$ is in G_i and (by induction) whether the elements of $\bar{r}$ are in G^*. To conclude that t is in G^* then, we simply needs to check that $r(\bar{v})$ is not equivalent in E to a variable and that no two distinct elements in $\bar{r}$ are equivalent in E. But this is possible because E is non-trivial and the word problem in E is decidable. □

We now show that every $(\Sigma_1 \cup \Sigma_2)$-term can be effectively reduced to an E-equivalent term in $T(\Sigma, G^*)$. To do that we will appeal to the correctness of a slight modification of the combination procedure of Section 4. The only significant change in the new procedure, shown in Figure 5, is that its input is an abstraction system instead of a pair of terms. Essentially in the same way as in Section 4.3, one can show that the new procedure is correct in the following sense:

Proposition 5.6 *The procedure in Figure 5 terminates for all inputs* S *and succeeds iff* S *is unsatisfiable in* E.

The following property of the procedure is also an immediate consequence of the results proved in Section 4.3.

[25] Unless $\Sigma = \Sigma_1 \cap \Sigma_2$ is empty, in which case $\bar{r}$ consists of the *distinct* alien subterms of t and $r(\bar{v})$ is obtained from $\widehat{t}$ by identifying any two variables of $\widehat{t}$ that abstract identical alien subterms.

[26] The different decompositions are generated by the fact that each Σ-symbol can be seen as either a Σ_1- or a Σ_2-symbol.

Lemma 5.7 *The final set S_n generated by the procedure on some input S_0 is an abstraction system. Furthermore,*

$$E \models \exists \bar{v}_0.S_0 \leftrightarrow \exists \bar{v}_n.S_n$$

where $\bar{v}_j$ is a sequence consisting of the left-hand side variables of S_j, for $j \in \{0, n\}$.

We have seen that, from every disequation $s \not\equiv t$ with $s, t \in T(\Sigma_1 \cup \Sigma_2, V)$, it is possible to produce an equivalent abstraction system. Specifically, one can use the purification procedure described in Subsection 4.1 to produce a system S such that

$$E \quad \models \quad (s \not\equiv t) \leftrightarrow \exists \bar{y}.S, \tag{3}$$

where $\bar{y}$ are the left-hand side variables of S. An inverse sort of process is also possible: given an abstraction system S, one can produce a disequation $s \not\equiv t$ such that (3) above holds.

In fact, if $S = \{x \not\equiv y\} \cup T$ is an abstraction system, the relation $\prec$ on T is acyclic. This means that its transitive closure $\prec^+$ is a strict partial ordering on the finite set T, and so it can be extended to a strict total ordering $<$ on T. Let

$$v_1 \equiv t_1 < v_2 \equiv t_2 < \cdots < v_k \equiv t_k$$

be the enumeration of T along this total ordering. We define θ_S to be the substitution obtained by the composition[27]

$$[v_1/t_1][v_2/t_2] \cdots [v_k/t_k].$$

In the following, we will call θ_S the *substitution induced by S*.

Lemma 5.8 *Let S be the abstraction system above and $\bar{v}$ a sequence consisting of the left-hand side variables of S. Then, $E \models (x\theta_S \not\equiv y\theta_S) \leftrightarrow \exists \bar{v}.S$.*

Proof. Recall that $\theta_S := [v_1/t_1][v_2/t_2] \cdots [v_k/t_k]$ where v_i does not occur in t_j for all $i \in \{1, \ldots, k\}$ and $j \geq i$. Then the claim is an easy consequence of the fact that

$$E \quad \models \quad (\exists v.(\varphi \wedge v \equiv t)) \leftrightarrow \varphi[v/t]$$

for every formula φ, term t, and variable v not occurring in t. □

It is useful to notice that, for all $v_i \equiv t_i \in S$, the term $t_i\theta_S$ is obtained essentially by "plugging in" into t_i all the terms $t_j\theta_S$ such that $v_j \equiv t_j \in S$ and $v_i \equiv t_i \prec v_j \equiv t_j$.

[27] Note that θ_S does not depend on which total extension of $\prec^+$ we take.

Lemma 5.9 *Let S_n be the final abstraction system generated by the procedure in Figure 5 on some input S. Where θ_{S_n} is the substitution induced by S_n, let*

$$S'_n \;:=\; \{v \equiv t\theta_{S_n} \mid v \equiv t \in S_n\}.$$

Then, the following holds for all $i = 1, 2$ and $u \equiv r\theta_{S_n}, v \equiv t\theta_{S_n} \in S'_n$ such that $u \equiv r, v \equiv t \in S_n$ and $r, t \in T(\Sigma_i, V)$.

1. *$r\theta_{S_n} \neq_E v$ for all $v \in V$;*
2. *if the height of $u \equiv r$ in S_n is > 0, then $r\theta_{S_n} \in G^*$;*
3. *if $u \neq v$, then $r\theta_{S_n} \neq_E t\theta_{S_n}$.*

Proof. Let $i \in \{1, 2\}$, $u \equiv t, v \equiv t \in S_n$ such that $r, t \in T(\Sigma_i, V)$ and $u \neq v$. We prove the claims simultaneously by induction on the ranks of $r\theta_{S_n}$ and $t\theta_{S_n}$.

(Base case) Assume that $r\theta_{S_n}$ and $t\theta_{S_n}$ have both rank 0. It is easy to see that then $r\theta_{S_n} = r$ and $t\theta_{S_n} = t$. To see that Point 1 holds, it is enough to observe that the only variables r can be equivalent to in E are its own. But if that were the case, either **Coll1** or **Coll2** would apply to S_n, against the assumption that S_n is a final abstraction system.

Now, if the height of $u \equiv r$ in S_n is > 0, u must occur in a right-hand side term of S_n. But then we can argue as in Lemma 4.14 that r is in G_i. Since we already know that r is not equivalent in E to any variable, that proves Point 2.

To prove Point 3 simply observe that, since both r and t are Σ_i-terms and E is a conservative extension of E_i (by Proposition 5.2), $r =_E t$ would imply $r =_{E_i} t$. But this is impossible because otherwise the **Ident** rule would apply to S_n, again against the assumption that S_n is a final abstraction system.

(Induction Step) Assume that $r\theta_{S_n}$ has rank greater than 0. Where $\bar{v}$ is the tuple consisting of the variables of r and $\bar{r} := \bar{v}\theta_{S_n}$, it is immediate that $r\theta_{S_n}$ has the form $r(\bar{r})$ where $\bar{r}$ contains no repetitions. Moreover, each non-variable term r_k in $\bar{r}$ is a right-hand side of S'_n of the form $r_k = t'\theta_{S_n}$ with $t' \in (T(\Sigma_1, V) \cup T(\Sigma_2, V)) \setminus T(\Sigma_i, V)$, which means that $\widehat{r_k} \notin T(\Sigma_i, V)$. Also note that the corresponding equation $v' \equiv t'$ in S_n has height > 0 since $(u \equiv r) \prec (v' \equiv t')$. Thus, by induction hypothesis, the terms in $\bar{r}$ are elements of G^*. Moreover, they are all pairwise inequivalent in E. In fact, a variable and a non-variable term of $\bar{r}$ are inequivalent by Point 1, pairs of non-variable terms are inequivalent by Point 3, and pairs of variable terms are inequivalent because E is non-trivial. As in the base case, we can show that, if the height of $u \equiv r$ in S_n is > 0, r is an element of $G_i \setminus V$ and is not equivalent in E to any variable. It follows then by Definition 5.4 that $r(\bar{r})$ is an element of G^*, which proves Point 2.

To prove Point 3 let $r' := r\theta_{S_n}$, $t' := t\theta_{S_n}$, and notice that $r' = u\theta_{S_n}$ and $t' = v\theta_{S_n}$ by construction of θ_{S_n}. Then consider the abstraction system T obtained from S_n

by replacing its disequation by $u \not\equiv v$. Since the equational part of T coincides with the one of S_n, we have that $\theta_T = \theta_{S_n}$, and so $r' = u\theta_T$ and $t' = v\theta_T$. By Lemma 5.8 then, $r' \not\equiv t'$ is satisfiable in E iff T is satisfiable in E. We know that no derivation rules apply to T, otherwise they would apply to S_n, which is impossible. Since u and v are distinct by assumption, we can then conclude that the procedure in Figure 5 fails on input T. By Proposition 5.6 then, T, and so $r' \not\equiv t'$, is satisfiable in E, which then entails that $r' \neq_E t'$.

Point 1 can be proven similarly to Point 3 by considering, for any variable v of $r\theta_{S_n}$, the abstraction system obtained from S_n by replacing its disequation by $u \not\equiv v$. Again, the argument is based on the fact that v is distinct from u, which this time is a consequence of the fact that $\prec$ is acyclic over S_n. Also note that the acyclicity of S_n and the definition of θ_{S_n} imply that $v\theta_{S_n} = v$ for all variables v occurring in $r\theta_{S_n}$. □

We can now show that, given any term in $T(\Sigma_1 \cup \Sigma_2, V)$, it is possible to find an equivalent term in $T(\Sigma, G^*)$.

Proposition 5.10 *For every term $t \in T(\Sigma_1 \cup \Sigma_2, V)$, there is a term $t' \in T(\Sigma, G^*)$, effectively computable from t, such that $t =_E t'$.*

Proof. Since $V \subseteq G^*$ by construction, it is enough to consider the case in which t is non-variable. Hence, assume that $t \in T(\Sigma_1 \cup \Sigma_2, V) \setminus V$.

Let v be a variable not in $\mathcal{V}ar(t)$, and S_n the final abstraction system generated by the procedure in Fig. 5 on input $S_0 := AS(v \not\equiv t)$. Then, let $t_n := y\theta_{S_n}$, where $x \not\equiv y$ is the disequation of S_n and θ_{S_n} the substitution induced by S_n. We first show that $t =_E t_n$.

By construction, S_0 has the form $\{v \not\equiv u\} \cup T$ with v not occurring in T. From the definition of the derivation rules used by the procedure it is easy to see that v is never replaced by other variables, which means that the disequation of S_n is in fact $v \not\equiv y$ and that $v\theta_{S_n} = v$. Then, by Proposition 4.3, Lemma 5.7, and Lemma 5.8 above it follows that the formulae:

$$(v \not\equiv t) \leftrightarrow \exists \bar{v}_0.S_0, \quad \exists \bar{v}_0.S_0 \leftrightarrow \exists \bar{v}_n.S_n, \quad \exists \bar{v}_n.S_n \leftrightarrow (v \not\equiv t_n),$$

where $\bar{v}_j$ are the left-hand side variables of S_j for $j \in \{0, n\}$, are all valid in E. This entails that $E \models (v \equiv t) \leftrightarrow (v \equiv t_n)$, from which it follows that $t =_E t_n$.

Now notice that S_n has the form $\{v \neq y, y \equiv r\} \cup R$ where $r \in T(\Sigma_i, V)$ for $i = 1$ or $i = 2$, and that $t_n = r\theta_{S_n}$.

Let $s(\bar{r}) = NF^{\Sigma}_{E_i}(r)$ and $t' := s(\bar{r})\theta_{S_n}$. Because of our assumption on $NF^{\Sigma}_{E_i}$, we know that no non-variable term in $\bar{r}$ is equivalent in E_i, and so in E, to a variable. Now, since $r =_{E_i} s(\bar{r})$, it is immediate that $t_n = r\theta_{S_n} =_E s(\bar{r})\theta_{S_n} = t'$ and thus $t =_E t'$. It is also immediate that t' is effectively computable from t_n, which was in turn computed from t. To prove the claim then it is enough to show that $t' \in T(\Sigma, G^*)$.

By an application of Lemma 5.9 we can show that $v\theta_{S_n} \in G^*$ for all $v \in \mathcal{V}ar(r)$. Recalling that $s(\bar{r}) \in T(\Sigma, G_i)$ and assuming as usual that the variables of $s(\bar{r})$ not occurring in t are all fresh, we can then show that $r'\theta_{S_n}$ is in G^* for every element r' of $\bar{r}$. That is immediate if r' is a variable. If r' is not a variable, let $\bar{v}'$ be its variables and $\bar{r}' := \bar{v}'\theta_{S_n}$. Using Lemma 5.9 again, it is not difficult to see that $r'(\bar{v}')$ and $\bar{r}'$ satisfy all the conditions in Definition 5.4, which means that $r'(\bar{r}') = r'\theta_{S_n}$ is in G^*. It follows that $t' = s(\bar{r}\theta_{S_n})$ is an element of $T(\Sigma, G^*)$. □

From what we have seen so far, G^* satisfies the first two requirements in Definition 3.6 for G^* to be a Σ-base of E. To show that it satisfies the third, we will use the following result about the model $\mathcal{A}$ of E constructed earlier as a fusion of the countably infinitely generated E_i-free algebras $\mathcal{A}_i$ ($i = 1, 2$).

Lemma 5.11 *Where $\mathcal{A}$ is the algebra given in Lemma 5.1, let α be an arbitrary bijective valuation of V onto Z_2. Then, for all $i = 1, 2$ and all $r, t \in G^* \setminus V$ with $\widehat{r}, \widehat{t} \in T(\Sigma_i, V)$,*

1. $[\![r]\!]^{\mathcal{A}}_{\alpha} \in Z_{2,i}$
2. $r =_E t$ *if* $[\![r]\!]^{\mathcal{A}}_{\alpha} = [\![t]\!]^{\mathcal{A}}_{\alpha}$.

Proof. We prove both claims simultaneously by induction on the rank of the terms.

(Base case) To prove Point 1 above, first assume that $r \in G_2 \cap (G^* \setminus V)$. We start by showing that $[\![r]\!]^{\mathcal{A}}_{\alpha} \in Y_2$. Since α is a bijective valuation of V onto Z_2, α_2 is a bijective valuation of V onto X_2, and $Z_2 \subseteq X_2$, there is a term r' obtained by a bijective renaming of the variables in r such that $[\![r]\!]^{\mathcal{A}}_{\alpha} = [\![r']\!]^{\mathcal{A}_2}_{\alpha_2}$. Since G_2 is closed under renaming, we have that $r' \in G_2$, and thus $[\![r']\!]^{\mathcal{A}_2}_{\alpha_2} \in Y_2$ by definition of Y_2. Now we prove by contradiction that $[\![r]\!]^{\mathcal{A}}_{\alpha} \notin X_2$. In fact, if $[\![r]\!]^{\mathcal{A}}_{\alpha} \in X_2$, it is easy to show that there is a $v \in V$ and an injective valuation γ of $\mathcal{V}ar(v \equiv r)$ into X_2 such that $\mathcal{A}, \gamma \models v \equiv r$. Recalling that $\mathcal{A}^{\Sigma_2}$ is free in E_2 over X_2 we then obtain by Proposition 2.1 that $v =_{E_2} r$, against the fact that $v \neq_E r$ by construction of G^* (see Definition 5.4(2a)). It follows that $[\![r]\!]^{\mathcal{A}}_{\alpha} \in Z_{2,2} = Y_2 \setminus X_2$.

Now assume that $r(\bar{v}) \in G_1 \cap (G^* \setminus V)$. Again, first we show that $[\![r]\!]^{\mathcal{A}}_{\alpha} \in Y_2$. Let $\beta_1 := h \circ \alpha_1$, as in Lemma 5.1. Since α is a bijective valuation of V onto Z_2, β_1 is a bijective valuation of V onto X'_1, and $Z_2 \subseteq X'_1$, there is a term r' obtained by a bijective renaming of the variables in r such that $[\![r]\!]^{\mathcal{A}}_{\alpha} = [\![r']\!]^{\mathcal{A}_1}_{\alpha_1}$. Again, $r' \in G_1$ as G_1 is closed under renaming, and thus $[\![r']\!]^{\mathcal{A}_1}_{\beta_1} \in Y_2$ by Lemma 5.1. As in the previous case, using the fact that $\mathcal{A}^{\Sigma_1}$ is free in E_1 over X'_1, we can prove $[\![r]\!]^{\mathcal{A}}_{\alpha} \notin X'_1$. It follows that $[\![r]\!]^{\mathcal{A}}_{\alpha} \in Z_{2,1} = Y_2 \setminus X'_1$.

To prove Point 2, let $i \in \{1,2\}$ and consider the terms $r, t \in G_i \cap (G^* \setminus V)$ such that $[\![r]\!]^{\mathcal{A}}_{\alpha} = [\![t]\!]^{\mathcal{A}}_{\alpha}$. Since both r, t are Σ_i-terms, this means that $\mathcal{A}^{\Sigma_i}, \alpha \models r \equiv t$. Now, observe that Z_2 is included in the set of generators of the free model $\mathcal{A}^{\Sigma_i}$ of E_i and that by construction α is an injection of $\mathcal{V}ar(r \equiv t)$ into Z_2. It follows by Proposition 2.1 that $r =_{E_i} t$, and so $r =_E t$.

(Induction step) Let $i, \iota \in \{1,2\}$ with $i \neq \iota$, and consider two terms t_1, t_2 in G^*, but not in $G_1 \cup G_2$, such that $\widehat{r}, \widehat{t} \in T(\Sigma_i, V)$. We know that, for $j = 1, 2$, t_j has the form

$$r_j(\bar{v}_j, \bar{r}_j)$$

where $r_j \in G_i \setminus V$, $\bar{v}_j \subseteq V$, $\bar{r}_j \subseteq G^* \setminus V$, $\bar{r}_j$ is nonempty, and $\widehat{r'} \in T(\Sigma_\iota, V)$ for all $r' \in \bar{r}_j$. Let $\bar{b}_j$ be the tuple of values that α assigns, in order, to the variables in $\bar{v}_j$, and $\bar{c}_j$ the tuple consisting, in order, of all the elements $[\![r']\!]^{\mathcal{A}}_{\alpha}$ with $r' \in \bar{r}_j$.

To prove Point 1, first notice that $\bar{b}_j \subseteq Z_2$ by definition of α and $\bar{c}_j \subseteq Z_{2,\iota}$ by induction hypothesis. It is immediate that $\bar{b}_j$ contains no repetitions and has no elements in common with $\bar{c}_j$. We claim that $\bar{c}_j$ contains no repetitions either. In fact, if $[\![r']\!]^{\mathcal{A}}_{\alpha} = [\![r'']\!]^{\mathcal{A}}_{\alpha}$ for two distinct $r', r'' \in \bar{r}_j$, we know by induction hypothesis that $r' =_E r''$. But this contradicts the fact that the tuple $\bar{r}_j$ must satisfy Definition 5.4(2d). Given these facts, noticing that $\mathcal{A}^{\Sigma_i}$ is free in E_i over $Z_{2,\iota} \cup Z_2$, it is easy to show (similarly to the base case) that $[\![r_j(\bar{v}_j, \bar{r}_j)]\!]^{\mathcal{A}}_{\alpha} = r_j^{\mathcal{A}^{\Sigma_i}}(\bar{b}_j, \bar{c}_j) \in Z_{2,i}$.

To prove Point 2, assume that $[\![t_1]\!]^{\mathcal{A}}_{\alpha} = [\![t_2]\!]^{\mathcal{A}}_{\alpha}$, which is to say that,

$$\mathcal{A}, \alpha \models r_1(\bar{v}_1, \bar{r}_1) \equiv r_2(\bar{v}_2, \bar{r}_2).$$

Let $\bar{u}_1, \bar{u}_2$ be tuples of variables abstracting $\bar{r}_1, \bar{r}_2$ in the equation above so that E-equivalent terms are abstracted by the same variable. From the proof of Point 1 above, it is easy to see that there is an injective valuation β into $Z_{2,\iota} \cup Z_2$ such that

$$\mathcal{A}, \beta \models r_1(\bar{v}_1, \bar{u}_1) \equiv r_2(\bar{v}_2, \bar{u}_2).$$

Since $r_1(\bar{v}_1, \bar{u}_1), r_2(\bar{v}_2, \bar{u}_2)$ are both Σ_i-terms and $\mathcal{A}^{\Sigma_i}$ is free in E_i over $Z_{2,\iota} \cup Z_2$, we can conclude that $r_1(\bar{v}_1, \bar{u}_1) =_{E_i} r_2(\bar{v}_2, \bar{u}_2)$, and so $r_1(\bar{v}_1, \bar{u}_1) =_E r_2(\bar{v}_2, \bar{u}_2)$. From this it is immediate that

$$t_1 = r_1(\bar{v}_1, \bar{r}_1) =_E r_2(\bar{v}_2, \bar{r}_2) = t_2$$

as well. □

We are now ready to prove that Σ is a set of constructors for E as well.

Proposition 5.12 *G^* is a Σ-base of E.*

Proof. We show that G^*, E, and Σ satisfy Definition 3.6.

Now, Definition 3.6(1) is a consequence of the definition of G^*, whereas Definition 3.6(2) holds by Proposition 5.10. To prove Definition 3.6(3) consider again the fusion $\mathcal{A}$ defined earlier and a bijection α of V onto Z_2.

Let $s_1(\bar{r}_1), s_2(\bar{r}_2)$ be terms in $T(\Sigma, G^*)$ and $s_1(\bar{v}_1), s_2(\bar{v}_2)$ the terms obtained from them by abstracting E-equivalent terms in $\bar{r}_1, \bar{r}_2$ with the same variable. Clearly $s_1(\bar{v}_1) =_E s_2(\bar{v}_2)$ implies $s_1(\bar{r}_1) =_E s_2(\bar{r}_2)$. Therefore, suppose that $s_1(\bar{r}_1) =_E s_2(\bar{r}_2)$. Since $\mathcal{A}$ is a model of E, $s_1(\bar{r}_1) =_E s_2(\bar{r}_2)$ entails that

$$\mathcal{A}, \alpha \models s_1(\bar{r}_1) \equiv s_2(\bar{r}_2).$$

Recall that $\mathcal{A}^\Sigma$ is free in E^Σ over $Y_2 = Z_{2,1} \cup Z_2 \cup Z_{2,2}$ and notice that, by Lemma 5.11, $[\![r]\!]^{\mathcal{F}}_\alpha \in Y_2$ for all $r \in G^*$. From this it is again easy to see that there is an injective valuation β of the variables of $\bar{v}_1, \bar{v}_2$ into the generators of $\mathcal{A}^\Sigma$ such that $\mathcal{A}^\Sigma, \beta \models s_1(\bar{v}_1) \equiv s_2(\bar{v}_2)$. It follows by Proposition 2.1 that $s_1(\bar{v}_1) =_{E^\Sigma} s_2(\bar{v}_2)$, which implies immediately that $s_1(\bar{v}_1) =_E s_2(\bar{v}_2)$. □

To sum up, we have obtained the following nice modularity result:

Theorem 5.13 *Let E_1, E_2 be two non-trivial equational theories with respective signatures Σ_1, Σ_2 such that, for $i = 1, 2$*

- *$\Sigma := \Sigma_1 \cap \Sigma_2$ is a set of constructors for E_i;*
- *${E_1}^\Sigma = {E_2}^\Sigma$;*
- *E_i admits a Σ-base G_i closed under bijective renaming of V;*
- *G_i is recursive and G_i-normal forms are computable for Σ and E_i;*
- *the word problem for E_i is decidable.*

Then the following holds:

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

Proof. Point 1 holds by Proposition 5.12 and Theorem 3.7; Point 2 holds by Corollary 5.3; Point 3 holds by Proposition 5.12 and the definition of G^*; given Point 3, Point 4 holds by Proposition 5.5 and Proposition 5.10; finally, Point 5 holds by Theorem 4.16; □

Because of its complete modularity, the above result extends immediately by iteration to the combination of more than two theories, all pairwise sharing the same set of constructors Σ and having the same Σ-restriction.

6 Conclusion and Open Questions

In this report, we have shown that the collapse-freeness requirement imposed on the definition of constructors given in [BT99] is not necessary for the modularity of the results given there on the combination of decision procedures for the word problem. To do this we have described and proved correct a variant of the combination procedure in [BT99], which can be used to combine decision procedures for the word problems for equational theories sharing non-collapse-free constructors. This broadens considerably the scope of our combination method and makes it in principle a viable tool for the study of combination results for modal logics as well.

Even with their already broader scope, the results presented here are preliminary in two respects. First, they themselves depend on two new *technical restrictions* for which we do not yet know whether they are necessary. One is the restriction in the definition of constructors that $X \subseteq Y$, and the other is the restriction in the combination procedure that the Σ-bases employed there be closed under renaming. In all the cases we have considered so far, the two restrictions are either immediately satisfied or can be assumed to be satisfied with no loss of generality. We are trying to find out whether they can then be removed altogether, or whether there is a fundamental (non-technical) reason for them.

Second, it is still unclear to what extent our combination procedure can help in the combination of modal logics. 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 particular modal logics in question.

References

[BN98] Franz Baader and Paliath Narendran. Unification of concept terms in description logics. In H. Prade, editor, *Proceedings of the 13th European Conference on Artificial Intelligence (ECAI-98)*, pages 331–335. John Wiley & Sons Ltd, 1998.

[BS98] Franz Baader and Klaus U. Schulz. Combination of constraint solvers for free and quasi-free structures. *Theoretical Computer Science*, 192:107–161, 1998.

[BT99] Franz Baader and Cesare Tinelli. Deciding the word problem in the union of equational theories sharing constructors. In P. Narendran and M. Rusinowitch, editors, *Proceedings of the 10th International Conference on Rewriting Techniques and Applications (Trento, Italy)*, volume 1631 of *Lecture Notes in Computer Science*, pages 175–189. Springer-Verlag, 1999. (longer version available as Technical Report UIUCDCS-R-98-2073, Department of Computer Science, University of Illinois at Urbana-Champaign, October 1998).

[DKR94] Eric Domenjoud, Francis Klay, and Christophe Ringeissen. Combination techniques for non-disjoint equational theories. In A. Bundy, editor, *Proceedings 12th International Conference on Automated Deduction, Nancy (France)*, volume 814 of *Lecture Notes in Artificial Intelligence*, pages 267–281. Springer-Verlag, 1994.

[DM79] Nachum Dershowitz and Zohar Manna. Proving termination with multiset orderings. *Communications of the ACM*, 22(8):465–476, August 1979.

[Fit93] Melvin Fitting. Basic modal logic. In D. M. Gabbay, C. J. Hogger, and J. A. Robinson, editors, *Handbook of Logic in Artificial Intelligence and Logic Programming, Vol. 1.* Oxford University Press, Oxford, UK, 1993.

[Gol76] Robert Goldblatt. Metamathematics of modal logics. *Reports on Mathematical Logic*, 6:41–78, 1976.

[Hem94] Edith Hemaspaandra. Complexity transfer for modal logic (extended abstract). In *Proceedings, Ninth Annual IEEE Symposium on Logic in Computer Science*, pages 164–173, Paris, France, 1994. IEEE Computer Society Press.

[Hod93] Wilfrid Hodges. *Model Theory*, volume 42 of *Enclyclopedia of mathematics and its applications.* Cambridge University Press, 1993.

[Kop88] Sabine Koppelberg. General theory of Boolean algebras. In J. Monk, editor, *Handbook of Boolean Algebras.* North-Holland, Amsterdam, The Netherlands, 1988.

[Kra99] Marcus Kracht. *Tools and Techniques in Modal Logic.* Elsevier, Amsterdam, The Netherlands, 1999.

[KW97] Marcus Kracht and Frank Wolter. Simulation and transfer results in modal logic: A survey. *Studia Logica*, 59:149–177, 1997.

[Lem66] Edward J. Lemmon. Algebraic semantics for modal logics. *J. of Symbolic Logic*, 31(1):46–65, 1966.

[Pig74] Don Pigozzi. The join of equational theories. *Colloquium Mathematicum*, 30(1):15–25, 1974.

[TR98] Cesare Tinelli and Christophe Ringeissen. Non-disjoint unions of theories and combinations of satisfiability procedures: First results. Technical Report UIUCDCS-R-98-2044, Department of Computer Science, University of Illinois at Urbana-Champaign, April 1998. (also available as INRIA research report no. RR-3402).