# Order-Sorted Rewriting and Congruence Closure

José Meseguer

Department of Computer Science
University of Illinois at Urbana-Champaign

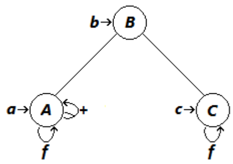
**Abstract.** Order-sorted type systems supporting inheritance hierarchies and subtype polymorphism are used in theorem proving, AI, and declarative programming. The satisfiability problems for the theories of: (i) order-sorted uninterpreted function symbols, and (ii) of such symbols *modulo* a subset $\Delta$ of associative-commutative ones are *reduced* to the *unsorted* versions of such problems at no extra computational cost. New results on order-sorted rewriting are needed to achieve this reduction.
**Keywords**: order-sorted rewriting, congruence closure, satisfiability.

## 1 Introduction

For greater expressiveness and efficiency, type systems supporting inheritance hierarchies and subtype polymorphism are used in many areas such as resolution theorem proving, e.g., [26,22], declarative logic and rule-based languages, e.g., [24,12,11,5], and artificial intelligence, e.g., [24,10]. Order-sorted (OS) equational logic, e.g., [16,20], is a logical framework supporting inheritance hierarchies and subtype polymorphism widely used for these purposes. Therefore, the development of *decision procedures* for OS theories is of interest in all these areas. However, except for, e.g., [13,6,25] this matter seems to have received relatively little attention. I focus here on decision procedures for the OS theory of *uninterpreted function symbols*, which in an unsorted setting is decided by congruence closure algorithms [23,21,8]. However, for greater expressiveness one can allow some of the function symbols, say in a subsignature $\Delta \subseteq \Sigma$, to be *interpreted* by some axioms $B_\Delta$. For example, for an unsorted subsignature $\Delta \subseteq \Sigma$ of binary function symbols, Bachmair, Tiwari and Vigneron [2] have given a congruence closure algorithm *modulo* the axioms $AC_\Delta$, asserting the associativity and commutativity of all symbols in $\Delta$. Therefore, I also study satisfiability in the OS theory $(\Sigma, AC_\Delta)$ of *uninterpreted function symbols* $\Sigma$ *modulo* $AC_\Delta$.

The most obvious approach would be to develop an *order-sorted* congruence closure algorithm along the lines of [13] and then extended it to the modulo $AC$ case. However, the main, somewhat surprising message of this paper is that such OS congruence closure algorithms *are not needed at all*: the already existing and efficient *unsorted* congruence closure algorithms in [23,21,8] and congruence closure modulo $AC_\Delta$ in [2] and tools supporting them can be reused *without change* and *at no extra cost* to solve the corresponding OS satisfiability problems.

**A Simple Example.** Consider the following order-sorted signature $\Sigma$




with sorts $A, B, C$, subsorts $A, C < B$, $f$ subsort-polymorphic with typings $f : A \to A$ and $f : C \to C$, and a binary $+$ with typing $+ : A\,A \to A$. Its so-called *theory of uninterpreted function symbols* is just the order-sorted equational theory $(\Sigma, \emptyset)$ with empty set of equations, whose class of models, $\mathbf{OSAlg}_{\Sigma}$, is that of *all* order-sorted $\Sigma$-algebras detailed in Section 2. Is the formula

$$(\flat) \qquad a = b \wedge b = c \wedge f(f(a)) = f(a) \wedge a + f(f(a)) \neq f(a) + a$$

$(\Sigma, \emptyset)$-*satisfiable*? The standard way to answer this question if $\Sigma$ were unsorted would be to: (1) compute the congruence closure of the first three equations; and (2) test the last inequality using such a congruence closure. Since, as pointed out in [17,2], unsorted congruence closure algorithms are *ground Knuth-Bendix* completion algorithms [19], an obvious way to try to answer this question would be to try to *complete* the first three equations into an equivalent set of confluent and terminating rewrite rules. But this runs into serious trouble. An order-sorted Knuth-Bendix completion algorithm such as [14] will orient $a = b$ and $b = c$ as $b \to a$ and $b \to c$ because rules must be *sort-decreasing*, i.e., rewrite to a term of equal or lower sort. This then generates the critical pair $a = c$, which is *unorientable*, so completion fails. Notice also that *replacement of equals by equals* does not hold in an order-sorted setting: from $a = b$ we *cannot* derive $f(a) = f(b)$, because $f(b)$ doesn't type. These difficulties were clearly felt by the authors of [13], the only order-sorted congruence closure algorithm I am aware of, which is quite complex and is *not* a Knuth-Bendix completion. They say:

> *An approach using rewriting* [...] *fails due to the well-known problem that rewriting with order-sorted rewrite rules may create ill-typed terms.*

Let us now widen the problem into one of *satisfiability modulo AC* by making the $+$ symbol associative-commutative. That is, we consider the axioms $AC_+ = \{x + y = y + x, (x + y) + z = x + (y + z)\}$, with $x, y, z$ of sort $A$, and ask: is the formula $(\flat)$ $(\Sigma, AC_+)$-*satisfiable*? For this case, I am not aware of any order-sorted $AC$-congruence closure algorithm, but un unsorted one based on ground $AC$-completion exists [2]. The trouble, again, is that *order-sorted* $AC$-completion as in [14] fails miserably in the *same* way ($a = c$ cannot be oriented).

**Wouldn't it be nice** if we could *completely ignore* all sort information in the above two OS satisfiability problems and solve them as *unsorted* problems using standard (and efficient!) congruence closure [23,21,8] and congruence closure modulo AC [2] algorithms? If this reduction method were *sound*, we could easily settle the $(\Sigma, \emptyset)$- and $(\Sigma, AC_+)$-satisfiability of $(\flat)$: the rules $R = \{a \to b, c \to$

$b, f(f(b)) \to f(b)\}$ are confluent and terminating and therefore a congruence closure for the first three equations. They are also an $AC_+$-congruence closure. Since the disequality $a + f(f(a)) \neq f(a) + a$ reduces to $b + f(b) \neq f(b) + b$, the formula ($\flat$) is $(\Sigma, \emptyset)$-satisfiable. However, since $b + f(b) =_{AC_+} f(b) + b$, ($\flat$) is $(\Sigma, AC_+)$-*unsatisfiable*. But is this *reduction* to unsorted satisfiability *sound*?

**Initial Algebra Semantics to the Rescue!** Ignoring the sort information of an OS signature $\Sigma$ is captured by a signature map $u : \Sigma \ni (f : s_1 \dots s_n \to s) \mapsto (f : U \overset{n}{\dots} U \to U) \in \Sigma^u$, where $U$ is the single "universe" sort in the unsorted signature $\Sigma^u$. As further detailed at the end of Section 2, $u$ induces a *reduct* map of algebras in the opposite direction, $\_|_u : \mathbf{Alg}_{\Sigma^u} \ni A \mapsto A|_u \in \mathbf{OSAlg}_{\Sigma}$, making each unsorted algebra $A$ into and order-sorted one $A|_u$, and such that for $E$ a set of ground OS $\Sigma$-equations we have the equivalence: $A|_u \models E \;\Leftrightarrow\; A \models E$. In particular, the $E$-initial unsorted $\Sigma^u$-algebra $T_{\Sigma^u/E}$ is mapped to the OS $\Sigma$-algebra $T_{\Sigma^u/E}|_u$ and, since $T_{\Sigma^u/E}|_u \models E$, there is a unique OS homomorphism $h : T_{\Sigma/E} \to T_{\Sigma^u/E}|_u$ from the $E$-initial OS $\Sigma$-algebra $T_{\Sigma/E}$.

But the poof of Theorem 5 shows that, for equations $E$ and disequations $D$, the conjunction $\bigwedge E \wedge \bigwedge D$ is satisfiable iff $T_{\Sigma(C)/E} \models \bigwedge E \wedge \bigwedge D$, where the variables $C$ of $E \cup D$ are seen as *fresh new constants* added to $\Sigma$ to get a supersignature $\Sigma(C) \supseteq \Sigma$, so that $\bigwedge E \wedge \bigwedge D$ becomes a *ground* formula. This gives us, in model-theoretic terms, the key to verify the soundness of the hoped-for *reduction* of the satisfiability for the theory of OS uninterpreted function symbols to that of the unsorted theory of uninterpreted function symbols: this reduction method will be *sound* if and only if the OS homomorphism $h : T_{\Sigma(C)/E} \to T_{\Sigma^u(C)/E}|_u$ is *injective.* In proof-theoretic terms this injectivity will hold if and only if for all ground $\Sigma$-equation $u = v$ we have the equivalence: $(\Sigma, E) \vdash u = v \;\Leftrightarrow\; (\Sigma^u, E) \vdash u = v$. The ($\Rightarrow$) direction is obvious, but the ($\Leftarrow$) direction is a non-trivial new result that follows from several *conservativity theorems* that I prove in Sections 3.2 and 4.1 by factorizign the signature map $u : \Sigma \to \Sigma^u$ through a sequence $\Sigma \hookrightarrow \Sigma^\square \to \widehat{\Sigma} \to \Sigma^u$ of increasingly simpler order-sorted, many-sorted and finally unsorted signatures and relating equational and rewriting deductions at all these levels.

**The Plot Thickens.** The soundness of the hoped-for reduction to the unsorted case is considerably thornier for satisfiability modulo $AC_\Delta$. As before, the reduction will be sound if and only if for ground $\Sigma$-equations $E$ the unique $\Sigma$-homomorphism $h : T_{\Sigma/E \cup AC_\Delta} \to T_{\Sigma^u/E \cup AC_{\Delta^u}}|_u$ from the initial $E \cup AC_\Delta$-algebra $T_{\Sigma/E \cup AC_\Delta}$ is *injective.* But some of the conservativity theorems along the above sequence of signature maps $\Sigma \hookrightarrow \Sigma^\square \to \widehat{\Sigma} \to \Sigma^u$ needed to make $h$ injective actually *break down* in the $AC_\Delta$ case. The problem has to do with the translation of the equations $AC_\Delta$ along these signature maps. At the unsorted level of $\Sigma^u$ the translated equations $AC_{\Delta^u}$, are *more general* and therefore *identify more terms* than the original OS equations $AC_\Delta$. Consider a simple example: the equation $a + b = b + a$ does not type in our example signature $\Sigma$, but it types in the supersignature $\Sigma^\square \supseteq \Sigma$, which for our running example is depicted in Section 3.1. The AC equations $AC_\Delta$ in our example are just associativity and commutativity of $+ : A\ A \to A$ and therefore *apply only* to terms of sort $A$. Instead, the AC equations $AC_{\Delta^u}$ are unsorted, and *apply to all terms*. This means that $a + b =_{AC_{\Delta^u}} b + a$, but since $b$ does not have sort $A$, we have $a + b \neq_{AC_\Delta} b + a$. It also means that the homomorphism $h' : T_{\Sigma^\square/E\cup AC_\Delta} \to T_{\Sigma^u/E\cup AC_{\Delta^u}}|_u$ in general is *not* injective. However, all hope is not lost. As a direct consequence of Corollary 2 in Section 3.2, there is an isomorphism $\alpha : T_{\Sigma/E\cup AC_\Delta} \cong T_{\Sigma^\square/E\cup AC_\Delta}|_\Sigma$ to the $\Sigma$-*reduct* of $T_{\Sigma^\square/E\cup AC_\Delta}$ and this shows that the homomorphism $h : T_{\Sigma/E\cup AC_\Delta} \to T_{\Sigma^u/E\cup AC_{\Delta^u}}|_u$ that we need to prove injective for the reduction to be sound is up to isomorphism a *restriction* of $h'$ to $T_{\Sigma/E\cup AC_\Delta}$, which *could* be injective even if $h'$ is not. Lemma 3 in Section 4.1 and the highly non-trivial Theorem 8 in Section 5 save the day: it follows from them that $h$ is indeed injective and the reduction is also sound for the $AC$ case. To the best of my knowledge the results on reducing order-sorted to unsorted satisfiability and on order-sorted rewriting and equality are new.

## 2 Preliminaries on Order-Sorted Algebra

The following material is adapted from [20], which generalizes [16]. It summarizes the basic notions of order-sorted algebra needed in the rest of the paper. It assumes the notions of many-sorted signature and many-sorted algebra, e.g., [9].

**Definition 1.** *An* order-sorted (OS) signature *is a triple* $\Sigma = (S, \leq, \Sigma)$ *with* $(S, \leq)$ *a poset and* $(S, \Sigma)$ *a many-sorted signature.* $\widehat{S} = S/{\equiv_\leq}$*, the quotient of* $S$ *under the equivalence relation* $\equiv_\leq = (\leq \cup \geq)^+$*, is called the set of* connected components *of* $(S, \leq)$*. The order* $\leq$ *and equivalence* $\equiv_\leq$ *are extended to sequences of same length in the usual way, e.g.,* $s'_1 \ldots s'_n \leq s_1 \ldots s_n$ *iff* $s'_i \leq s_i$, $1 \leq i \leq n$. $\Sigma$ *is called* sensible *if for any two* $f : w \to s, f : w' \to s' \in \Sigma$*, with* $w$ *and* $w'$ *of same length, we have* $w \equiv_\leq w' \;\Rightarrow\; s \equiv_\leq s'$*. A* many-sorted signature $\Sigma$ *is the special case where the poset* $(S, \leq)$ *is discrete, i.e.,* $s \leq s'$ *iff* $s = s'$.

*For connected components* $[s_1], \ldots, [s_n], [s] \in \widehat{S}$

$$f_{[s]}^{[s_1]\ldots[s_n]} = \{f : s'_1 \ldots s'_n \to s' \mid s'_i \in [s_i]\ \ 1 \leq i \leq n,\ s' \in [s]\}$$

*denotes the family of "subsort polymorphic" operators* $f$. $\square$

**Definition 2.** *For* $\Sigma = (S, \leq, \Sigma)$ *an OS signature, an* order-sorted $\Sigma$-algebra $A$ *is a many-sorted* $(S, \Sigma)$*-algebra* $A$ *such that:*

– *whenever* $s \leq s'$*, then we have* $A_s \subseteq A_{s'}$*, and*
– *whenever* $f : w \to s, f : w' \to s'$ *in* $f_{[s]}^{[s_1]\ldots[s_n]}$ *(with* $w = s_1 \ldots s_n$*), and* $\overline{a} \in A^w \cap A^{w'}$*, then we have* $A_{f:w\to s}(\overline{a}) = A_{f:w'\to s'}(\overline{a})$.

*An* order-sorted $\Sigma$-homomorphism $h : A \to B$ *is a many-sorted* $(S, \Sigma)$*-homomorphism such that whenever* $[s] = [s']$ *and* $a \in A_s \cap A_{s'}$*, then we have* $h_s(a) = h_{s'}(a)$. $h$ *is* injective*, resp.* surjective*, resp.* bijective*, iff for each* $s \in S$ $h_s$ *is injective, resp. surjective, resp. bijective. We call* $h$ *an* isomorphism *if there* *is another order-sorted $\Sigma$-homomorphism $g : B \to A$ such that for each $s \in S$, $h_s; g_s = 1_{A_s}$, and $g_s; h_s = 1_{B_s}$, with $1_{A_s}, 1_{B_s}$ the identity functions on $A_s, B_s$. This defines a category $\mathbf{OSAlg}_\Sigma$.* $\square$

**Theorem 1.** *[20] The category $\mathbf{OSAlg}_\Sigma$ has an initial algebra. Furthermore, if $\Sigma$ is sensible, then the term algebra $T_\Sigma$ with:*

- *if $a : \lambda \to s$ then $a \in T_{\Sigma,s}$,*
- *if $t \in T_{\Sigma,s}$ and $s \leq s'$ then $t \in T_{\Sigma,s'}$,*
- *if $f : s_1 \dots s_n \to s$ and $t_i \in T_{\Sigma,s_i}$ $1 \leq i \leq n$, then $f(t_1, \dots, t_n) \in T_{\Sigma,s}$,*

*is initial, i.e., has a unique $\Sigma$-homomorphism to each $\Sigma$-algebra.*

For $[s] \in \widehat{S}$, $T_{\Sigma,[s]}$ denotes the set $T_{\Sigma,[s]} = \bigcup_{s' \in [s]} T_{\Sigma,s'}$. Similarly, $T_\Sigma$ will (ambiguously) denote both the above-defined $S$-sorted set and the set $T_\Sigma = \bigcup_{s \in S} T_{\Sigma,s}$. We say that an OS signature $\Sigma$ *has non-empty sorts* iff for each $s \in S$, $T_{\Sigma,s} \neq \emptyset$. *We will assume throughout that* $\Sigma$ has non-empty sorts.

An $S$-sorted set $X = \{X_s\}_{s \in S}$ of *variables*, satisfies $s \neq s' \Rightarrow X_s \cap X_{s'} = \emptyset$, and the variables $X$ are always assumed disjoint from all constants in $\Sigma$. The $\Sigma$-*term algebra* on variables $X$, $T_\Sigma(X)$, is the *initial algebra* for the signature $\Sigma(X)$ obtained by adding to $\Sigma$ the variables $X$ *as extra constants*. Since a $\Sigma(X)$-algebra is just a pair $(A, \alpha)$, with $A$ a $\Sigma$-algebra, and $\alpha$ an *interpretation of the constants* in $X$, i.e., an $S$-sorted function $\alpha \in [X \to A]$, the $\Sigma(X)$-initiality of $T_\Sigma(X)$ can be expressed as the following corollary of Theorem 1:

**Theorem 2.** *(Freeness Theorem). If $\Sigma$ is sensible, for each $A \in \mathbf{OSAlg}_\Sigma$, $\alpha \in [X \to A]$ there exists a unique $\Sigma$-homomorphim, denoted $\_\alpha : T_\Sigma(X) \longrightarrow A$, such that for each $s \in S$, and each $x \in X_s$ we have $x\alpha_s = \alpha_s(x)$.*

The first-order language of *equational $\Sigma$-formulas*[1] is defined in the usual way: its atoms are $\Sigma$-*equations* $t = t'$, where $t, t' \in T_\Sigma(X)_{[s]}$ for some $[s] \in \widehat{S}$ and each $X_s$ is assumed countably infinite. The set *Form*$(\Sigma)$ of *equational $\Sigma$-formulas* is then inductively built from atoms by: conjunction ($\wedge$), disjunction ($\vee$) negation ($\neg$), and universal ($\forall x{:}s$) and existential ($\exists x{:}s$) quantification with sorted variables $x{:}s \in X_s$ for some $s \in S$. The literal $\neg(t = t')$ is denoted $t \neq t'$.

The *satisfaction* relation between $\Sigma$-algebras and formulas is defined in the usual way: given a $\Sigma$-algebra $A$, a formula $\varphi \in \mathit{Form}(\Sigma)$, and an assignment $\alpha \in [Y \to A]$, with $Y = \mathit{fvars}(\varphi)$ the free variables of $\varphi$, we define the satisfaction relation $A, \alpha \models \varphi$ inductively as usual: for atoms, $A, \alpha \models t = t'$ iff $t\alpha = t'\alpha$; for Boolean connectives it is the corresponding Boolean combination of the satisfaction relations for subformulas; and for quantifiers: $A, \alpha \models (\forall x{:}s)\ \varphi$ (resp. $A, \alpha \models (\exists x{:}s)\ \varphi$) holds iff for all $a \in A_s$ (resp. there is an $a \in A_s$) we have $A, \alpha \uplus \{(x{:}s, a)\} \models \varphi$, where the assignment $\alpha \uplus \{(x{:}s, a)\}$ extends $\alpha$ by mapping $x{:}s$ to $a$. Finally, $A \models \varphi$ holds iff $A, \alpha \models \varphi$ holds for each $\alpha \in [Y \to A]$, where

[1] There is only an apparent lack of predicate symbols. To express a predicate $p(x_1{:} s_1, \dots, x_n{:}s_n)$, add a new sort *Truth* with a constant *tt*, and with $\{\mathit{Truth}\}$ a separate connected component, and view $p$ as a function symbol $p : s_1, \dots, s_n \to \mathit{Truth}$. An atomic formula $p(t_1, \dots, t_n)$ is then expressed as the equation $p(t_1, \dots, t_n) = tt$.

$Y = \mathit{fvars}(\varphi)$. We say that $\varphi$ is *valid* (or *true*) in $A$ iff $A \models \varphi$. We say that $\varphi$ is *satisfiable* in $A$ iff $\exists\alpha \in [Y \to A]$ such that $A, \alpha \models \varphi$, where $Y = \mathit{fvars}(\varphi)$.

An *order-sorted equational theory* is a pair $T = (\Sigma, E)$, with $E$ a set of $\Sigma$-equations. $\mathbf{OSAlg}_{(\Sigma,E)}$ denotes the full subcategory of $\mathbf{OSAlg}_{\Sigma}$ with objects those $A \in \mathbf{OSAlg}_{\Sigma}$ such that $A \models E$, called the $(\Sigma, E)$-*algebras*. $\mathbf{OSAlg}_{(\Sigma,E)}$ has an *initial algebra* $T_{\Sigma/E}$ [20], further discussed in Section 3. Given $T = (\Sigma, E)$ and $\varphi \in \mathit{Form}(\Sigma)$, we call $\varphi$ $T$-*valid*, written $E \models \varphi$, iff $A \models \varphi$ for each $A \in \mathbf{OSAlg}_{(\Sigma,E)}$. We call $\varphi$ $T$-*satisfiable* iff there exists $A \in \mathbf{OSAlg}_{(\Sigma,E)}$ with $\varphi$ satisfiable in A. Note that $\varphi$ is $T$-*valid* iff $\neg\varphi$ is $T$-*unsatisfiable*.

$\Sigma = ((S, \leq), \Sigma)$ is a *subsignature* of $\Sigma' = ((S', \leq'), \Sigma')$, denoted $\Sigma \subseteq \Sigma'$, iff $(S, \leq) \subseteq (S', \leq')$ is a subposet inclusion, and $\Sigma \subseteq \Sigma'$. A *signature map* $H : \Sigma \to \Sigma'$ is a monotonic function $H : (S, \leq) \to (S', \leq')$ of the underlying posets of sorts together with a mapping $H : \Sigma \ni (f : s_1 \ldots s_n \to s) \mapsto (H(f) : H(s_1) \ldots H(s_n) \to H(s)) \in \Sigma'$. $H$ induces a map $H : \mathit{Form}(\Sigma) \to \mathit{Form}(\Sigma')$. A signature inclusion $\Sigma \subseteq \Sigma'$ is a simple signature map $\Sigma \hookrightarrow \Sigma' : f \mapsto f$.

A signature map $H : \Sigma \to \Sigma'$ induces a functor in the *opposite* direction $\_\,|_H$: $\mathbf{OSAlg}_{\Sigma'} \ni B \mapsto B\,|_H \in \mathbf{OSAlg}_{\Sigma}$, where the $H$-*reduct* $B\,|_H$ has: (i) for each $s \in S$, $(B\,|_H)_s = B_{H(s)}$; and (ii) for each $f : s_1 \ldots s_n \to s$ in $\Sigma$, $(B\,|_H)_f = B_{H(f)}$. For $H : \Sigma \hookrightarrow \Sigma'$ a signature inclusion, $B\,|_H$ is denoted $B\,|_{\Sigma}$. For $B \in \mathbf{OSAlg}_{\Sigma'}$ and $\varphi \in \mathit{Form}(\Sigma)$ with $\mathit{fvars}(\varphi) = \emptyset$ we have [20]:

$$(\dagger) \quad B \models H(\varphi) \;\Leftrightarrow\; B\,|_H \models \varphi.$$

## 3 Order-Sorted Rewriting and Equality

Given an OS signature $\Sigma = ((S, \leq), \Sigma)$, a $\Sigma$-*rewrite rule*[2] is a sequent $l \to r$ with $l, r \in T_{\Sigma}(X)_{[s]}$ for some $[s] \in \widehat{S}$. An *order-sorted term rewriting system* (OSTRS) is then a pair $(\Sigma, R)$ with $R$ a set of $\Sigma$-rewrite rules.

Since, as shown in the Introduction, replacement of equals for equals and standard rewriting break down in the order-sorted case, we should define rewriting deductions with an OSTRS not by means of the reflexive-transitive closure $\to_R^*$ of the rewrite relation $\to_R$, but by means of an *inference system* with two kinds of *sequents*: sequents $t \to t'$, where $t, t' \in T_{\Sigma}(X)_{[s]}$, $[s] \in \widehat{S}$, corresponding to *one-step* application of rules, and sequents $t \to^{\circledast} t'$, where $t, t' \in T_{\Sigma}(X)_{[s]}$, $[s] \in \widehat{S}$, corresponding to more complex rewriting deductions. The symbol $\to^{\circledast}$ is close enough to $\to^*$ to suggest that: (i) it plays a role similar to a reflexive transitive-closure in the unsorted case, but (ii) in general it is *different* for such a closure. For example, for $\Sigma$ the signature in the Introduction and $R = \{a \to b, b \to c\}$, we can derive $f(a) \to^{\circledast} f(c)$, but there is no sequence of one-step rewrites from $f(a)$ to $f(c)$. We then define two kinds of *rewriting deductions*: $(\Sigma, R) \vdash t \to t'$ and $(\Sigma, R) \vdash t \to^{\circledast} t'$, as those sequents derivable from $(\Sigma, R)$ by a finite application of the following inference rules, where $\sigma$ denotes an $S$-sorted *substitution*, i.e., an $S$-sorted function $\sigma \in [X \to T_{\Sigma}(X)]$:

[2] For greater generality no restriction is placed on the variables of $l$ and $r$.

**Reflexivity**
$$\frac{}{t \to^{\circledast} t}$$

**Subsumption**
$$\frac{t \to t'}{t \to^{\circledast} t'}$$

**Transitivity**
$$\frac{t \to^{\circledast} t' \qquad t' \to^{\circledast} t''}{t \to^{\circledast} t''}$$

**Congruence**
$$\frac{u_1 \to^{\circledast} u_1' \quad \ldots \quad u_n \to^{\circledast} u_n'}{f(u_1, \ldots, u_n) \to^{\circledast} f(u_1', \ldots, u_n')}$$
$$\text{where } f(u_1, \ldots, u_n), f(u_1', \ldots, u_n') \in T_{\Sigma}(X)$$

**Replacement**
$$\frac{}{t\sigma \to t'\sigma}$$
$$\text{where } t \to t' \in R$$

The first three and the last inference rule are standard, but the **Congruence** rule is more subtle. We can better understand these rules by means of our running example $(\Sigma, R)$. The sequent $f(a) \to^{\circledast} f(b)$ is *not* derivable: the attempt to obtain it by applying **Replacement** with rule $a \to b$, **Subsumption** to get $a \to^{\circledast} b$, and then **Congruence** fails, because of the side condition, since $f(b) \notin T_{\Sigma}(X)$. To see what *can* be derived, consider the derivation of the sequent $f(a) \to^{\circledast} f(c)$. Since we have rules $a \to b$ and $b \to c$, we can obviously derive $a \to^{\circledast} c$ by two applications of **Replacement** followed by **Subsumption** and one application of **Transitivity**. Then **Congruence** gives us:

$$\frac{a \to^{\circledast} c}{f(a) \to^{\circledast} f(c)}$$

Note the interesting fact that $f(a)$ is typed with $f : A \to A$, and $f(c)$ is typed with $f : C \to C$. We can think of **Congruence** as a "tunneling rule." $f(a) \to^{\circledast} f(c)$ *cannot* be obtained by composing one-step rewrites: failed attempts such as that for deriving $f(a) \to^{\circledast} f(b)$ make it impossible; but we can "tunnel through" such failed attempts and obtain a more complex sequent like $f(a) \to^{\circledast} f(c)$ when the left- and right-hand sides are well-formed terms in $T_{\Sigma}(X)$.

The above inference system yields as a *special case* a sound and complete inference system for *order-sorted equational logic*: we just view an order-sorted equational theory $(\Sigma, E)$ as the OSTRS $(\Sigma, R(E))$, where $R(E) = \{t \to t' \mid t = t' \in E \;\; \vee \;\; t' = t \in E\}$. That is, equality steps are viewed as either left-to-right or right-to-left rewrite steps. We then have:

**Definition 3.** *Given an order-sorted equational theory $(\Sigma, E)$ with $\Sigma$ sensible, its* equational deduction relation, *denoted $(\Sigma, E) \vdash u = v$, or just $E \vdash u = v$, is defined by the equivalence:*

$$(\Sigma, E) \vdash u = v \quad \Leftrightarrow \quad (\Sigma, R(E)) \vdash u \to^{\circledast} v.$$

**Theorem 3.** *(Soundeness and Completeness). For $\Sigma$ sensible and $E \cup \{u = v\}$ a set of $\Sigma$-equations we have the equivalence:*

$$(\Sigma, E) \vdash u = v \quad \Leftrightarrow \quad (\Sigma, E) \models u = v$$

The above theorem has as a corollary the construction of the *initial algebra* $T_{\Sigma/E}$ for the category $\mathbf{OSAlg}_{(\Sigma,E)}$ of $(\Sigma,E)$-algebras. Assuming $\Sigma$ sensible, $T_{\Sigma/E}$, has an easy definition. Note that the relation $E \vdash u = v$ induces an equivalence relation $=_E$ on each set $T_{\Sigma,[s]}$, $[s] \in \widehat{S}$. We then define for each $s' \in [s]$ the set $T_{\Sigma/E,s'} = \{[t]_{=_E} \in T_{\Sigma,[s]} \mid [t]_{=_E} \cap T_{\Sigma,s'} \neq \emptyset\}$, and define each operation $f : s_1 \dots s_n \to s \in \Sigma$ by the map $([t_1]_{=_E}, \dots, [t_1]_{=_E}) \mapsto [f(t'_1, \dots, t'_n)]_{=_E}$, where $t'_i \in [t_i]_{=_E} \cap T_{\Sigma,s_i}$, $1 \leq i \leq n$, showing it does not depend on the choice of $t'_i$'s.

### 3.1 Kind-Complete OS-Rewriting and Equational Deduction

The order-sorted rewrite relation $t \rightarrow^{\circledast} t'$ is obviously quite impractical and hard to implement. For this reason, given an OSTRS $(\Sigma, R)$ several conditions on either $\Sigma$ or $R$ have been sought to be able to perform rewriting computations in essentially the standard and efficient way in which it is performed in an unsorted or many-sorted TRS. Two such conditions, going back to [15], are to either: (i) require that the rules $R$ are *sort-decreasing*, i.e., for each $l \to r \in R$, if $l\sigma \in T_{\Sigma,s}$ then $r\sigma \in T_{\Sigma,s}$; or (ii) if $R$ is not sort-decreasing, extend $\Sigma$ with new "retract operators" $r_{s,s'} : s \to s'$, $s, s' \in [s]$, $s \not\leq s'$, to catch typing errors, add to $R$ "error recovery" rules of the form $r_{s,s'}(x{:}s) \to x{:}s$, and force sort-decreasingness of $R$ by replacing each not sort-decreasing $u \to v \in R$ by suitable rules of the form $u\sigma \to r_{s,s'}(v\sigma)$, where $\sigma$ may lower the sorts of some variables.

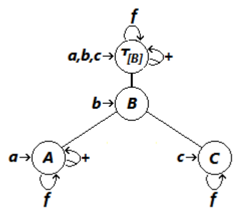
Conditions (i) or in its defect (ii) work and can be shown to be conservative in a certain sense [15]. However, they have serious limitations. Sort decreasingness is a strong condition that may be impossible to achieve for some OSTRS arising in practice. If the solution with retracts is adopted, an unpleasant consequence is that we *change the models*, including the initial ones, since retracts add new operations and new error terms *to the original sorts*.

All these limitations can be avoided —while allowing rewriting with rules $R$ and equational deduction with equations $E$ to be performed in the *standard* way— by using a *faithful embedding* of order-sorted equational logic into *membership equational logic* (MEL) [20,4]. MEL introduces a typing distinction between *sorts* $s \in S$, which may be related by subsort relations just as in the order-sorted way, and the *kind* $\top_{[s]}$ associated to each connected component $[s] \in \widehat{S}$, which is above all sorts in $[s]$. An ill-formed term like $f(b)$ in the OS signature of the Introduction has no sort, but has kind $\top_{[B]}$. In this way, the earlier side condition in the **Congruence** rule in Section 3 can be avoided.

That this embedding of logics is faithful means in particular that *both* initial models and equational deduction are preserved ([20], Corollary 28). However: (i) the proof in [20] is model-theoretic; (ii) it focuses on the equational logic level, and does not deal with the more general rewriting logic level; and (iii) it assumes that the entire MEL framework is adopted. Can the essential advantages of this embedding be still obtained *while remaining at the order-sorted level*? The answer is *yes!* Since: (i) this solution plays a key role in the treatment of satisfiability for the theory of OS uninterpreted function symbols in Section 4, and (ii) having a much simpler theory of OS rewriting is useful in its own right, I give a detailed treatment of it below. The key idea is to use a signature transformation $\Sigma \mapsto \Sigma^{\Box}$ extending any OS signature $\Sigma$ into one whose components have a top sort, understood as the kind of that component. The essential point is that $\Sigma^{\Box}$ belongs to a class of order-sorted signatures called *kind complete* where both rewriting and equational deduction can be performed in the standard way.

**Definition 4.** *An OS signature $\Sigma = ((S, \leq), \Sigma)$ is called* kind-complete *iff each connected component $[s] \in \widehat{S}$ has a top sort $\top_{[s]}$, called its* kind, *with $\top_{[s]} \geq s'$ for each $s' \in [s]$, and any subsort-polymorphic family $f^{[s_1]\ldots[s_n]}_{[s]} \subseteq \Sigma$ includes the typing $f : \top_{[s_1]}, \ldots, \top_{[s_n]} \to \top_{[s]}$. Note that any many-sorted $\Sigma$ —and in particular any unsorted (i.e., single-sorted) $\Sigma$— is trivially kind-complete.*

Any OS signature $\Sigma$ can be extended to a kind-complete one by a transformation $\Sigma \mapsto \Sigma^{\Box}$. $\Sigma^{\Box}$ is constructed in two-steps: (i) we first associate to the order-sorted signature $((S, \leq), \Sigma)$ the many-sorted signature $\widehat{\Sigma} = (\widehat{S}_{\top}, \widehat{\Sigma})$, where $\widehat{S}_{\top} = \{\top_{[s]} \mid [s] \in \widehat{S}\}$, and with $f : \top_{[s_1]} \ldots \top_{[s_n]} \to \top_{[s]} \in \widehat{\Sigma}$ iff $f^{[s_1]\ldots[s_n]}_{[s]} \subseteq \Sigma$; and (ii) we then define $\Sigma^{\Box} = ((S \uplus \widehat{S}_{\top}, \leq_{\Box}), \Sigma \cup \widehat{\Sigma})$, where $\leq_{\Box} \cap S^2 = \leq$, and for each $\top_{[s]} \in \widehat{S}_{\top}$ we have $s' < \top_{[s]}$ for each $s' \in [s]$. That is, we add $\top_{[s]}$ as a top sort above each $s' \in [s]$ and add the new typing $f : \top_{[s_1]} \ldots \top_{[s_n]} \to \top_{[s]}$ for each $f^{[s_1]\ldots[s_n]}_{[s]} \subseteq \Sigma$. For $\Sigma$ the signature in the Introduction, $\Sigma^{\Box}$ is as follows:




We then have subsignature inclusions: $\Sigma \subseteq \Sigma^{\Box}$ and $\widehat{\Sigma} \subseteq \Sigma^{\Box}$. Note that, by construction, if $\Sigma$ is sensible, both $\widehat{\Sigma}$ and $\Sigma^{\Box}$ are also sensible; and that the initial algebra $T_{\Sigma^{\Box}}$ is *preserved by reducts*, i.e., we have:

$$T_{\Sigma^{\Box}}|_{\Sigma} = T_{\Sigma} \quad \textit{and} \quad T_{\Sigma^{\Box}}|_{\widehat{\Sigma}} = T_{\widehat{\Sigma}}.$$

For kind-complete signatures, rewriting, and in particular equational deduction, can be performed in the standard way. Recall the usual notation to denote term positions, subterms, decompositions and term replacement from [7]: (i) positions in a term viewed as a tree are marked by strings $p \in \mathbb{N}^*$, (ii) $t|_p$ denotes the subterm of term $t$ at position $p$, (iii) $t = t[t|_p]_p$ denotes a *decomposition* of $t$ into a context $t[]_p$ and its subterm $t|_p$, and (iv) $t[u]_p$ denotes the result of *replacing* subterm $t|_p$ at position $p$ by $u$.

**Definition 5.** *Let $(\Sigma, R)$ be an OSTRS with $\Sigma$ sensible and kind-complete. The* one-step *R-rewrite relation $u \to_R v$, holds between $u, v \in T_{\Sigma}(X)_{[s]}$, $[s] \in \widehat{S}$, iff* *there is a rewrite rule* $t \to t' \in R$, *a substitution* $\sigma \in [X \to T_\Sigma(X)]$, *and a term position* $p$ *in* $u$ *such that* $u = u[t\sigma]_p$ *and* $v = u[t'\sigma]_p$.

*We denote by* $\to_R^+$ *the transitive closure of* $\to_R$, *and by* $\to_R^*$ *the reflexive-transitive closure of* $\to_R$, *and write* $(\Sigma, R) \vdash u \to_R^* v$ *to make* $\Sigma$ *explicit.*

$(\Sigma, R)$ *is called* terminating *iff* $\to_R$ *is a well-founded relation; and is called* confluent *iff whenever* $t \to_R^* u$ *and* $t \to_R^* v$ *there exists* $w$ *such that* $u \to_R^* w$ *and* $v \to_R^* w$. $(\Sigma, R)$ *is called* convergent *iff it is both confluent and terminating. If* $(\Sigma, R)$ *is convergent, each* $\Sigma$*-term* $t$ *rewrites by some* $t \to_R^* t!_R$ *to a unique term* $t!_R$, *called its* $R$-canonical form, *that cannot be further rewritten.*

Note that, since $\Sigma$ is kind-complete, if $u \in T_\Sigma(X)_{[s]}$, $t \to t' \in R$, and $u = u[t\sigma]_p \in T_\Sigma(X)_{[s]}$, then we always have $u[t'\sigma]_p \in T_\Sigma(X)_{[s]}$. That is, $\to_R$ *never produces ill-formed terms*, so that in the above definition of $\to_R$ the requirement the $v \in T_\Sigma(X)_{[s]}$ is unnecessary and does not have to be checked. Indeed, for kind-complete signatures order-sorted rewriting *becomes standard rewriting*:

**Lemma 1.** *Let* $(\Sigma, R)$ *be an OSTRS with* $\Sigma$ *sensible and kind-complete. Then we have the equivalence:*

$$(\Sigma, R) \vdash u \to^{\circledast} v \quad \Leftrightarrow \quad (\Sigma, R) \vdash u \to_R^* v.$$

**Corollary 1.** *Let* $\Sigma$ *be a sensible and kind-complete OS signature, and* $E \cup \{u = v\}$ *a set of* $\Sigma$*-equations. Then we have the equivalence:*

$$(\Sigma, E) \vdash u = v \quad \Leftrightarrow \quad (\Sigma, R(E)) \vdash u \to_{R(E)}^* v.$$

### 3.2 Conservativity Results

The whole point of the signature transformation $\Sigma \mapsto \Sigma^\square$ is to replace complex deductions of the form $(\Sigma, R) \vdash u \to^{\circledast} v$ by simple rewrite sequences $u \to_R^* v$ in the *extended* OSTRS $(\Sigma^\square, R)$. But is this sound?

**Theorem 4.** *Let* $(\Sigma, R)$ *be an OSTRS with* $\Sigma$ *sensible. Then for any* $u, v \in T_\Sigma(X)_{[s]}$, $[s] \in \widehat{S}$ *we have the equivalence:*

$$(\Sigma, R) \vdash u \to^{\circledast} v \quad \Leftrightarrow \quad (\Sigma^\square, R) \vdash u \to_R^* v.$$

**Corollary 2.** *Let* $\Sigma$ *be a sensible OS signature and* $E \cup \{u = v\}$ *a set of* $\Sigma$*-equations. Then we have the equivalences:*

$$(\Sigma, E) \vdash u = v \quad \Leftrightarrow \quad (\Sigma^\square, E) \vdash u = v \quad \Leftrightarrow \quad (\Sigma^\square, R(E)) \vdash u \to_{R(E)}^* v.$$

Since, besides the subsignature inclusion $\Sigma \subseteq \Sigma^\square$, we also have the inclusion $\widehat{\Sigma} \subseteq \Sigma^\square$, we have a further conservativity result:

**Lemma 2.** *Let* $\Sigma$ *be a sensible OS signature and* $(\widehat{\Sigma}, R)$ *a many-sorted TRS. Then for any* $u, v \in T_{\widehat{\Sigma}}(X)_{\top_{[s]}}$, $\top_{[s]} \in \widehat{S}_\top$, *where* $X = \{X_{\top[s]}\}_{\top_{[s]} \in \widehat{S}_\top}$, *we have* $(\widehat{\Sigma}, R) \vdash u \to_R^* v$ *iff* $(\Sigma^\square, R) \vdash u \to_R^* v$. *As an immediate consequence, for* $E \cup \{u = v\}$ *a set of* $\widehat{\Sigma}$*-equations, we have the equivalence:*

$$(\widehat{\Sigma}, E) \vdash u = v \quad \Leftrightarrow \quad (\Sigma^\square, E) \vdash u = v.$$

## 4 Order-Sorted $(\Sigma, \emptyset)$-QF-Satisfiability

In theorem proving the theory $(\Sigma, \emptyset)$, whose category of algebras is $\mathbf{OSAlg}_\Sigma$, is called the theory of *uninterpreted function symbols* $\Sigma$. As remarked in Definition 1, a *many-sorted* signature $\Sigma$ is a special case of an order-sorted signature, and an *unsorted* signature is a many-sorted signature where $S = \{U\}$ is a singleton set. Let $\mathit{QFForm}(\Sigma) \subseteq \mathit{Form}(\Sigma)$ denote the set of *quantifier-free* $\Sigma$-formulas, i.e., formulas with no quantifiers. When $\Sigma$ is unsorted, $(\Sigma, \emptyset)$-*QF-satisfiability*, i.e., $(\Sigma, \emptyset)$-satisfiability for any $\varphi \in \mathit{QFForm}(\Sigma)$ is *decidable* [1]. The goal of this section is to show that the same holds for any sensible OS signature $\Sigma$ by a *reduction* method. This can be done by two reductions. The first reduces this decidability problem to that of the *OS word problem*, which is the problem of whether, given a sensible OS signature $\Sigma$ and a finite set $E \cup \{u = v\}$ of *ground* $\Sigma$-equations, $E \vdash u = v$ holds or not. The desired first reduction is as follows:

**Theorem 5.** *$(\Sigma, \emptyset)$-QF-satisfiability is decidable for any sensible order-sorted signature $\Sigma$ iff the OS word problem is decidable.*

The proof follows from the more general Theorem 7 in Section 5, which deals with the OS word problem *modulo* equations $B$. The theorem's algorithmic content mirrors its proof: $\varphi = \bigvee_{1 \leq i \leq n}(\bigwedge E_i \wedge \bigwedge D_i)$ in DNF with the $E_i$ equalities and the $D_i$ disequalities is satisfiable iff, when we view the variables in $\varphi$ as fresh new constants $C$, there is an $i$, $1 \leq i \leq n$, such that $E_i \not\vdash u = v$ for each $u \neq v \in D_i$. Furthermore, $\bigwedge E_i \wedge \bigwedge D_i$ is satisfiable iff $T_{\Sigma(C)/E_i} \models \bigwedge E_i \wedge \bigwedge D_i$.

The second reduction is from the OS word problem to the *unsorted* word problem. This is broken into *two* reductions: (i) of the many-sorted word problem to the unsorted word problem in Section 4.1, and (ii) of the OS word problem to the many-sorted word problem in Section 4.2.

For $\Sigma$ *unsorted* and $E \cup \{u = v\}$ a finite set of ground $\Sigma$-equations it is well-known that the word problem $E \vdash u = v$ can be decided by a *congruence closure* algorithm [23,21,8]. What the various such algorithms have in common is that they are all instances (by applying difference strategies) of the same *abstract congruence closure* algorithm in the sense of [2], which is summarized below.

### 4.1 Abstract Congruence Closure

What the abstract congruence closure algorithm in [2] captures is what all concrete congruence closure algorithms have in common: they all are efficient, specialized *ground Knuth-Bendix* completion algorithms [19,17,2]: they all begin with a set $E$ of ground equations, and return a set $R$ of *convergent* ground rewrite rules $R$ equivalent to $E$ (on a possibly extended signature). We can then decide the word problem $E \vdash u = v$ by checking the syntactic equality $u!_R = v!_R$.

The key notion of *abstract congruence closure* in [2] is then as follows:

**Definition 6.** *[2] For $\Sigma$ an unsorted signature and $E$ a finite set of ground $\Sigma$-equations, an* abstract congruence closure *for $E$ is a set $R$ of ground convergent $\Sigma(K)$-rewrite rules, where $K$ is a finite set of new constants, such that: (i) they* *are either of the form* $c \to c'$*, with* $c, c' \in K$*, or of the form* $f(c_1, \ldots, c_n) \to c$*, with* $c_1, \ldots, c_n, c \in K$*,* $f \in \Sigma$ *with* $n \geq 0$ *arguments; (ii) for each* $c \in K$ *there is a ground* $\Sigma$*-term* $t$ *such that* $t!_R = c!_R$*; and (iii) for any ground* $\Sigma$*-equation* $u = v$ *we have* $E \vdash u = v$ *iff we have the syntactic equality* $u!_R = v!_R$*.*

The paper [2] then gives an *abstract congruence closure algorithm* described by six inference rules, with an optional seventh, such that: (i) takes as input a triple $(\emptyset, E, \emptyset)$ with $E$ is a set of ground $\Sigma$-equations; (ii) operates on triples of the form $(K', E', R')$ with $E'$ (resp. $R'$) the current $\Sigma(K')$-equations (resp. $\Sigma(K')$-rules); and (iii) terminates with a triple of the form $(K, \emptyset, R)$ such that $R$ is a congruence closure for $E$. The name *abstract congruence closure* is well-deserved: the algorithms in [23,21,8], and two other ones, are all shown to be *instantiations* of the abstract algorithm by applying the inference rules with different *strategies*, so that both the operation of each algorithm and its actual complexity are faithfully captured by the corresponding instantiation [2].

We need to decide *the many-sorted* word problem as a step for deciding the more general order-sorted one. But the many-sorted word problem can be easily *reduced* to the unsorted one by means of the signature transformation $\Sigma \ni (f : s_1 \ldots s_n \to s) \mapsto (f : U \overset{n}{\ldots} U \to U) \in \Sigma^u$, where $\Sigma = (S, \Sigma)$ is a many-sorted signature. Then all boils down to the following lemma:

**Lemma 3.** *For* $\Sigma$ *a sensible many-sorted signature and* $E$ *a set of* regular $\Sigma$-*equations —i.e.,* $t$ *and* $t'$ *have the same variables for each* $t = t' \in E$*— we have* $(\Sigma, E) \vdash u = v$ *iff* $(\Sigma^u, E^u) \vdash (u = v)^u$*, where for any* $\Sigma$*-equation* $t = t'$*,* $(t = t')^u$ *leaves the terms unchanged but regards all variables as unsorted.*

This lemma has a very practical consequence: we can use an unsorted congruence closure algorithm to solve the many-sorted word problem *at no extra cost*: no changes are needed either to the input $E$ or to the unsorted algorithm.

### 4.2 Deciding OS $(\Sigma, \emptyset)$-QF-Satisfiability

For any sensible OS signature $\Sigma$ we have reduced the decidability of the $(\Sigma, \emptyset)$-QF-satisfiability problem to that of the OS word problem in Theorem 5. And in Lemma 3 we have reduced the many-sorted word problem to the unsorted word problem, which is decidable by a congruence closure algorithm. To prove the decidability of the OS $(\Sigma, \emptyset)$-QF-satisfiability problem and obtain a correct algorithm for it we just need to reduce the OS word problem to the many-sorted word problem. For this, the conservativity results in Section 3.2 are crucial:

**Theorem 6.** *Let* $\Sigma$ *be a sensible OS signature and* $E \cup \{u = v\}$ *a set of ground* $\Sigma$*-equations. Then we have the equivalence:*

$$(\Sigma, E) \vdash u = v \quad \Leftrightarrow \quad (\widehat{\Sigma}, E) \vdash u = v.$$

The decidability of the OS $(\Sigma, \emptyset)$-QF-satisfiability problem goes back to [13]; but the reduction achieved by Theorem 5, Lemma 3 and Theorem 6 yields a new, very simple and efficient algorithm for deciding OS $(\Sigma, \emptyset)$-QF-satisfiability. Using a lazy $DPLL(\Sigma^u, \emptyset)$ solver (see, e.g., [3]), we do *not* have to assume that $\varphi$ is in DNF: after working on the Boolean abstraction of $\varphi$, the $DPLL(\Sigma^u, \emptyset)$ solver will ask questions about the satisfiability of formulas of the form: $\bigwedge E \ \wedge \ \bigwedge D$, where $E$ (resp. D) is a finite set of ground $\Sigma(C)$-equations (resp. $\Sigma(C)$-inequations). Satisfiability is then decided by:

1. regarding at no cost $\bigwedge E \ \wedge \ \bigwedge D$ as a ground $\Sigma(C)^u$-formula;
2. computing a congruence closure $R$ for $E$ in $O(|E| \ log(|E|))$; and
3. testing whether $u!_R \neq v!_R$ for each $u \neq v \in D$.

Therefore we can *reuse* the same algorithms and tools used in the *unsorted* case *at no extra cost*: the input to such algorithms and the algorithms or tools themselves need no changes, and the complexity is that of the unsorted case.

## 5 Order-Sorted $(\Sigma, AC_\Delta)$-QF-Satisfiability

Let $\Sigma$ be a sensible OS signature with $\Delta \subseteq \Sigma$ made exclusively of binary function symbols, say, $g, h, \ldots$, each of the form $g : s\,s \to s$ for some sorts $s \in S$, and with any typing of any such $g$ in $\Sigma$ necessarily a typing in $\Delta$, i.e., $\Delta$ and $(\Sigma - \Delta)$ share no symbols. Assume that each subsort-polymorphic family $g^{[s]\,[s]}_{[s]} \subseteq \Delta$ has always a biggest possible typing $g : s_g\, s_g \to s_g$ such that for any other typing $g : s\,s \to s$ in $g^{[s]\,[s]}_{[s]}$ we have $s \leq s_g$. We impose the *associativity-commutativity* (AC) of the subsort-polymorphic family $g^{[s]\,[s]}_{[s]}$ with the equations: $AC_g = \{g(x, y) = g(y, x), g(x, g(y, z)) = g(g(x, y), z)\}$ with $x, y, z$ of sort $s_g$. We furthermore require that the axioms $AC_g$ are *sort-preserving*, that is, that for each $S$-sorted substitution $\sigma$ and each sort $s \in S$ we have: $g(x, y)\sigma \in T_\Sigma(X)_s \ \Leftrightarrow$ $g(y, x)\sigma \in T_\Sigma(X)_s$, and $g(x, g(y, z))\sigma \in T_\Sigma(X)_s \ \Leftrightarrow \ g(g(x, y), z)\sigma \in T_\Sigma(X)_s$, which can be easily checked by the method explained in [18]. Let $AC_\Delta$ denote the set $AC_\Delta = \bigcup_{g \in \Delta} AC_g$ requiring all symbols in $\Delta$ to be $AC$. Call $(\Sigma, AC_\Delta)$ satisfying the above requirements the OS theory of $\Sigma$ *uninterpreted function symbols* $\Sigma$ *modulo* $AC_\Delta$. When $\Sigma = \Delta$ is unsorted and has a single symbol $+$, this is called the *theory of commutative semigroups.*

We can generalize the above setting by replacing $(\Delta, AC_\Delta)$ by any OS theory $(\Delta, B)$ with $\Delta$ sensible and considering any sensible supersignature $\Sigma \supseteq \Delta$ with $\Delta$ and $\Sigma - \Delta$ not sharing any symbols. Call $(\Sigma, B)$ the theory of *uninterpreted function symbols* $\Sigma$ *modulo* $B$. We can then reduce the decidability of the $(\Sigma, B)$-QF-satisfiabilty problem to that of the *OS word problem modulo* $B$, defined as the problem of whether given any $\Sigma \supseteq \Delta$ as above, and a set $E \cup \{u = v\}$ of ground $\Sigma$-equations, $E \cup B \vdash u = v$ holds or not. The reduction is as follows:

**Theorem 7.** *For any $(\Delta, B)$ and $\Sigma \supseteq \Delta$ as above, $(\Sigma, B)$-QF-satisfiability is decidable iff the OS word problem modulo $B$ is decidable.*

For $\Sigma \subseteq \Delta$ unsorted, Bachmair, Tiwari and Vigneron [2] have developed an *AC congruence closure* algorithm for the theory $(\Sigma, AC_\Delta)$ that decides the word problem modulo $AC_\Delta$ and therefore, by above Theorem 7, the unsorted $(\Sigma, AC_\Delta)$-QF-satisfiability problem. In the spirit of Section 4, the main goal of this section is to *reduce* the decidability of the OS $(\Sigma, AC_\Delta)$-QF-satisfiability problem to that of its unsorted version, and to furthermore *reuse* the *same* unsorted AC congruence closure algorithm in [2] to decide *at no extra cost* and with the *same complexity* the OS $(\Sigma, AC_\Delta)$-QF-satisfiability problem.

The decidability of OS $(\Sigma, AC_\Delta)$-QF-satisfiability has already been reduced to that of the OS word problem modulo $AC_\Delta$, now we just need to reduce the OS word problem modulo $AC_\Delta$ to the unsorted word problem modulo $AC_{\Delta^u}$.

This is achieved in two steps. First, we reduce the many-sorted word problem modulo $AC_{\widehat{\Delta}}$ to the unsorted word problem modulo $AC_{\Delta^u}$ using the $\widehat{\Sigma} \mapsto \Sigma^u$ transformation of Section 4.1. This first reduction is easy: the equations $AC_{\widehat{\Delta}}$ are *regular*. Therefore, if $E \cup \{u = v\}$ is a finite set of ground many-sorted $\widehat{\Sigma}$-equations, the equations $E \cup AC_{\widehat{\Delta}}$ are also regular and the conditions of Lemma 3 apply. We then reduce the OS word problem modulo $AC_\Delta$ to the many-sorted word problem modulo $AC_{\widehat{\Delta}}$. The $\widehat{\Delta}$-equations $AC_{\widehat{\Delta}}$ are obtained from the OS $\Delta$-equations in $AC_\Delta$ by replacing each variable $x{:}s$ by the variable $x{:}\top_{[s]}$. That is, for $E \cup \{u = v\}$ a finite set of *ground* $\Sigma$-equations must show the equivalence:

$$(\Sigma, E \cup AC_\Delta) \vdash u = v \quad \Leftrightarrow \quad (\widehat{\Sigma}, E \cup AC_{\widehat{\Delta}}) \vdash u = v$$

which, by Corollary 2, reduces to proving the equivalence:

$$(\Sigma^\Box, E \cup AC_\Delta) \vdash u = v \quad \Leftrightarrow \quad (\widehat{\Sigma}, E \cup AC_{\widehat{\Delta}}) \vdash u = v$$

which, by Lemma 2, follows as a special case from the more general theorem:

**Theorem 8.** *Let $\Sigma \supseteq \Delta$ be a sensible OS supersignature, $R$ a set of $\Sigma$-rewrite rules, and $u, v \in T_\Sigma(X)$. Then we have the equivalence:*

$$(\Sigma^\Box, R \cup AC_\Delta) u \rightarrow^*_{R \cup R(AC_\Delta)} v \quad \Leftrightarrow \quad (\Sigma^\Box, E \cup AC_{\widehat{\Delta}}) \vdash u \rightarrow^*_{R \cup R(AC_{\widehat{\Delta}})} v.$$

## 6 Related Work and Conclusions

[13] presents the only *order-sorted* congruence closure algorithm I am aware of. It provides a good solution under some extra assumptions on $\Sigma$, but it requires a quite complex congruence generation method and has worse complexity, $O(n^2)$, than the best $O(n\ log(n))$ unsorted algorithms. The papers [17,2] present the view of congruence closure as completion. In particular, the notions of *abstract congruence closure* and *AC-congruence closure* are due to [2]. The first study I know of satisfiability modulo theories in an order-sorted setting is [25].

The above-mentioned work has influenced and motivated the present one. The good news is that we get all the benefits of order-sorted $(\Sigma, \emptyset)$- and $(\Sigma, AC_\Delta)$-satisfiability *for free*, with no added computational cost and being able to reuse unsorted tools. At a more theoretical level, the order-sorted rewriting and equality results presented here are also good news and belong to the foundations of such an area. Future work will focus on exploiting these results at the tool level.

**Acknowledgements**. Partially supported by NSF Grant CNS 13-19109.

## References


1. Ackermann, W.: Solvable Cases of the Decision Problem. Noth-Holland (1954)
2. Bachmair, L., Tiwari, A., Vigneron, L.: Abstract congruence closure. J. Autom. Reasoning 31(2), 129–168 (2003)
3. Barrett, C., Tinelli, C.: Satisfiability modulo theories. In: Clarke, E., Henzinger, T., Veith, H. (eds.) Handbook of Model Checking. Springer (2014), (to appear)
4. Bouhoula, A., Jouannaud, J.P., Meseguer, J.: Specification and proof in membership equational logic. Theoretical Computer Science 236, 35–132 (2000)
5. Clavel, M., Durán, F., Eker, S., Meseguer, J., Lincoln, P., Martí-Oliet, N., Talcott, C.: All About Maude. Springer LNCS Vol. 4350 (2007)
6. Comon, H., Delor, C.: Equational formulae with membership constraints. Inf. Comput. 112(2), 167–216 (1994)
7. Dershowitz, N., Jouannaud, J.P.: Rewrite systems. In: van Leeuwen, J. (ed.) Handbook of Theoretical Computer Science, Vol. B, pp. 243–320. North-Holland (1990)
8. Downey, P.J., Sethi, R., Tarjan, R.E.: Variations on the common subexpressions problem. Journal of the ACM 27(4), 758–771 (1980)
9. Ehrig, H., Mahr, B.: Fundamentals of Algebraic Specification 1. Springer (1985)
10. Frisch, A.M.: The substitutional framework for sorted deduction: Fundamental results on hybrid reasoning. Artif. Intell. 49(1-3), 161–198 (1991)
11. Futatsugi, K., Diaconescu, R.: CafeOBJ Report. World Scientific (1998)
12. Futatsugi, K., Goguen, J., Jouannaud, J.P., Meseguer, J.: Principles of OBJ2. In: Proc. POPL 1985. pp. 52–66. ACM (1985)
13. Gallier, J., Isakowitz, T.: Order-sorted congruence closure. Tech. Rep. CIS-686, UPenn (1988), `http://repository.upenn.edu/cis_reports/686`
14. Gnaedig, I., Kirchner, C., Kirchner, H.: Equational completion in order-sorted algebras. Theor. Comput. Sci. 72(2&3), 169–202 (1990)
15. Goguen, J., Jouannaud, J.P., Meseguer, J.: Operational semantics of order-sorted algebra. In: Proc. ICALP 1985, vol. 194, pp. 221–231. Springer LNCS (1985)
16. Goguen, J., Meseguer, J.: Order-sorted algebra I. Theoretical Computer Science 105, 217–273 (1992)
17. Kapur, D.: Shostak's congruence closure as completion. In: Proc. RTA-97. vol. 1232, pp. 23–37. Springer LNCS (1997)
18. Kirchner, C., Kirchner, H., Meseguer, J.: Operational semantics of OBJ3. In: Proc. ICALP 1988, vol. 317, pp. 287–301. Springer LNCS (1988)
19. Knuth, D., Bendix, P.: Simple word problems in universal algebra. In: Leech, J. (ed.) Computational Problems in Abstract Algebra. Pergamon Press (1970)
20. Meseguer, J.: Membership algebra as a logical framework for equational specification. In: Proc. WADT'97. pp. 18–61. Springer LNCS 1376 (1998)
21. Nelson, G., Oppen, D.C.: Fast decision procedures based on congruence closure. J. ACM 27(2), 356–364 (Apr 1980)
22. Schmidt-Schauss, M.: Computational aspects of order-sorted logic with term declarations. Springer LNCS 395 (1989)
23. Shostak, R.E.: An algorithm for reasoning about equality. Communications of the ACM 21(7), 583–585 (Jul 1978)
24. Smolka, G., Aït-Kaci, H.: Inheritance hierarchies: Semantics and unification. J. Symb. Comput. 7(3/4), 343–370 (1989)
25. Tinelli, C., Zarba, C.G.: Combining decision procedures for sorted theories. In: Proc. JELIA 2004. vol. 3229, pp. 641–653. Springer LNCS (2004)
26. Walther, C.: A mechanical solution of Schubert's steamroller by many-sorted resolution. Artif. Intell. 26(2), 217–224 (1985)

## A Proofs of Theorems and Lemmas

**Proof of Theorem** 3

*Proof.* Since we only care about sequents of the form $u \to^{\circledast} v$, we can simplify the OSTRS inference system into an equivalent one for such sequents where **Replacement** deduces sequents of the form $t\sigma \to^{\circledast} t'\sigma$ and **Subsumption** is dropped. Identifying then $u \to^{\circledast} v$ with $u = v$ this system coincides with the order-sorted equational deduction inference system in Section 11 of [20], where the **Replacement** rule coincides with rule **Modus Ponens** there in the case, as assumed here, when the equations $E$ are *unconditional*. All other inference rules have the same name in both systems.

There are however three small differences: (i) the rules in Section 11 of [20] work on explicitly quantified equations, whereas the rewiting-based ones do not; this is because we have assumed that $\Sigma$ always has *non-empty sorts*, in which case such explicit quantification can be safely dropped; (ii) the rewriting-based inference system is missing the **Symmetry** rule; but that rule is unnecessary, since it is easy to show by structural induction that $(\Sigma, R(E)) \vdash u \to^{\circledast} v$ iff $(\Sigma, R(E)) \vdash v \to^{\circledast} u$; and (iii) the inference rules in Section 11 of [20] allow more general sets $\tilde{X}$ of variables, where $s \neq s' \Rightarrow X_s \cap X_{s'}$ need not hold; but this is inconsequential: assuming such a restriction throughout does not affect the derivable equations $u = v$ when $u, v \in T_\Sigma(X)$ and $X$ satisfies the restriction.

Since the OS equational inference rules in Section 11 of [20] are sound and complete (Theorem 24 there), the same holds for the present rewriting-based system. $\square$

**Proof of Lemma** 1

*Proof.* The proof that $u \to^{\circledast} v \;\Rightarrow\; u \to_R^* v$ is an easy structural induction on the structure of proofs for $u \to^{\circledast} v$. Because of the **Reflexivity** and **Transitivity** rules, to prove that $u \to_R^* v \;\Rightarrow\; u \to^{\circledast} v$ it is enough to prove that $u \to_R v \;\Rightarrow\; u \to^{\circledast} v$. But this follows by one application of **Replacement**, followed by **Subsumption**, followed by $|p|$ applications on **Congruence**, where $p$ is the position at which the rewriting $u \to_R v$ happens, and $|p|$ is the length of the string $p$. $\square$

**Proof of Theorem** 4

*Proof.* Since $\Sigma \subseteq \Sigma^\square$, obviously, $(\Sigma, R) \vdash u \to^{\circledast} v \;\Rightarrow\; (\Sigma^\square, R) \vdash u \to^{\circledast} v$. Therefore, Lemma 1 gives us the implication $(\Sigma, R) \vdash u \to^{\circledast} v \;\Rightarrow\; u \to_R^* v$. We just have to prove the other direction, i.e., that for any $u, v \in T_\Sigma(X)_{[s]}$, $[s] \in \widehat{S}$, $u \to_R^* v$ with OSTRS $(\Sigma^\square, R)$ implies $(\Sigma, R) \vdash u \to^{\circledast} v$. The proof is by contradiction. Suppose the implication does not hold. This means that the set of pairs $\{(u, n) \in T_\Sigma(X) \times \mathbb{N} \mid (\exists v \in T_\Sigma(X))\ (u \to_R^n v \wedge (\Sigma, R) \not\vdash u \to^{\circledast} v)\}$ is non-empty. Since we can define a lexicographic well-founded order $(u, n) > (u', m)$ on such pairs by the equivalence: $(u, n) > (u', m) \;\Leftrightarrow\; ht(u) > ht(u') \vee (ht(u) = ht(u') \;\wedge\; n > m)$, where $ht(u)$ is the height of $u$ as a tree, there is a minimal element, say $(u, n)$, under that order in the above set, and we must have $n > 0$ and a rewrite sequence:

$$u \to_R w_1 \to_R \dots w_{n-1} \to_R v$$

with $v \in T_\Sigma(X)$ and $(\Sigma, R) \not\vdash u \to^{\circledast} v$. Furthermore, we must have $w_i \notin T_\Sigma(X)$, $1 \leq i \leq n-1$, since otherwise the minimality of $(u, n)$ would be violated.

Now note that, since for $X = \{X_s\}_{s \in S}$ we have $T_{\Sigma^\square}(X)|_\Sigma = T_\Sigma(X)$ and therefore $T_{\Sigma^\square}(X)_s = T_\Sigma(X)_s$, $s \in S$, any $S$-sorted substitution $\sigma \in [X \to T_{\Sigma^\square}(X)]$ is actually an $S$-sorted substitution $\sigma \in [X \to T_\Sigma(X)]$. This means that for all rules $t \to t' \in R$ and all $\sigma \in [X \to T_{\Sigma^\square}(X)]$ we must have $t\sigma, t'\sigma \in T_\Sigma(X)$. But then this forces all positions $p_1, \dots, p_n$ at which the above $n$ rewrite steps take place to be different from the empty string. That is, $u$ and $v$ must be of the form $u = f(u_1, \dots, u_k)$, $v = f(v_1, \dots, v_k)$, $k > 1$, and we must have $u_j \to_R^* v_j$, $1 \leq j \leq k$. Furthermore, since we have $u_j, v_j \in T_\Sigma(X)$, $1 \leq j \leq k$, the minimality of $(u, n)$ forces $(\Sigma, R) \vdash u_j \to^{\circledast} v_j$, $1 \leq j \leq k$. But then the **Congruence** rule gives us $(\Sigma, R) \vdash u \to^{\circledast} v$, contradicting $(\Sigma, R) \not\vdash u \to^{\circledast} v$. $\square$

**Proof of Lemma** 2

*Proof.* The essential point is that for $X = \{X_{\top_{[s]}}\}_{\top_{[s]} \in \widehat{S}}$, we have $T_{\Sigma^\square}(X)|_{\widehat{\Sigma}} = T_{\widehat{\Sigma}}(X)$. Therefore, again for the same variables $X$, any well-sorted substitution $\sigma \in [X \to T_{\Sigma^\square}(X)]$ is also a well-sorted substitution $\sigma \in [X \to T_{\widehat{\Sigma}}(X)]$. This immediately gives us $(\widehat{\Sigma}, R) \vdash u \to_R^* v$ iff $(\Sigma^\square, R) \vdash u \to_R^* v$, as desired. $\square$

**Proof of Theorem** 5: see that of **Theorem** 7.

**Proof of Lemma** 3

*Proof.* Since many-sorted and unsorted signatures are kind-complete, we can use the $R(E)$ and $R(E^u)$ rewriting-based inference systems obtained by specializing to $\Sigma$ (resp. $\Sigma^u$) that in Corollary 1. An easy induction on the length of the rewrite sequence reduces everything to showing that for each $t \to t' \in R(E^u)$ and each $u \in T_\Sigma(X)_s$ and $v \in T_{\Sigma^u}(X)$ —where in the second case $X$ denotes the single-sorted set $X = \bigcup_{s \in S} X_s$ and all sort information is ignored— if $u \to_{R(E^u)} v$ is obtained by rewriting with $t \to t'$ at position $p$ in the unsorted signature $\Sigma^u$, we must have $v \in T_\Sigma(X)_s$ and $u \to_{R(E)} v$.

Let $u \in T_\Sigma(X)_s$ and suppose that there is an *unsorted* substitution $\sigma$ and a position $p$ such that $u = u[t\sigma]_p$ and we can apply rule $t \to t' \in R(E^u)$ to perform a rewrite step $u \to_{R(E^u)} u[t'\sigma]_p$. Since all equations in $E$ are regular, if $x{:}s'$ is a variable in $t \to t' \in R(E)$, then it belongs to both $t$ and $t'$. Since $u|_p = t\sigma$ is a $\Sigma$-term and $x{:}s'$ occurs in $t$, $\sigma(x{:}s')$ is a $\Sigma$-term. Furthermore, since $\Sigma$ sensible implies that $s' \neq s'' \Rightarrow T_\Sigma(X)_{s'} \cap T_\Sigma(X)_{s''} = \emptyset$, the $\Sigma$-term $\sigma(x{:}s')$ can *only* have sort $s'$, in spite of the fact that $\sigma$ was unsorted and disregarded sorts. Therefore, $\sigma$ is actually an $S$-sorted substitution for the variables of $t \to t'$. Therefore, $t'\sigma$ is also a $\Sigma$-term, $u[t'\sigma]_p \in T_\Sigma(X)_s$ and $u \to_{R(E)} u[t'\sigma]_p$, as desired. $\square$

**Proof of Theorem** 6

*Proof.* By Corollary 2 we have the equivalence:

$$(\Sigma, E) \vdash u = v \quad \Leftrightarrow \quad (\Sigma^{\square}, E) \vdash u = v.$$

But, since the equations $E \cup \{u = v\}$ are gound, they are trivially $\widehat{\Sigma}$-equations, so that Lemma 2 gives us the equivalence:

$$(\Sigma^{\square}, E) \vdash u = v \quad \Leftrightarrow \quad (\widehat{\Sigma}, E) \vdash u = v.$$

Stringing these two equivalences together we get our desired equivalence:

$$(\Sigma, E) \vdash u = v \quad \Leftrightarrow \quad (\widehat{\Sigma}, E) \vdash u = v.$$

□

**Proof of Theorem** 7

*Proof.* Let us first prove a lemma:

**Lemma 4.** *Let $\Sigma$ be sensible, and $B \cup E \cup G$ be $\Sigma$-equations with $E \cup G$ a finite set of ground equations. The following are equivalent:*

1. $E \cup B \not\vdash u = v$ *for each* $u = v \in G$
2. $T_{\Sigma/E \cup B} \models u \neq v$ *for each* $u = v \in G$
3. $\bigwedge E \wedge \bigwedge_{u=v \in G} u \neq v$ *is* $(\Sigma, B)$*-satisfiable.*

*Proof.* (1) $\Leftrightarrow$ (2) follows directly from the definition of $T_{\Sigma/E \cup B}$, and (2) $\Rightarrow$ (3) is trivial. We just need to prove (3) $\Rightarrow$ (2). But (3) just means that there is a $(\Sigma, E \cup B)$-algebra $A$ such that for each $u = v \in G$ $h([u]) \neq h([v])$, where $h : T_{\Sigma/E \cup B} \to A$ is the unique $\Sigma$-homomorphism guaranteed by the initiality of $T_{\Sigma/E \cup B}$, which forces $[u] \neq [v]$ and therefore $T_{\Sigma/E \cup B} \models u \neq v$ for each $u = v \in G$, as desired. □

The ($\Rightarrow$) implication is now trivial, since Lemma 4 shows that $E \cup B \vdash u = v$ holds iff $\bigwedge E \wedge u \neq v$ is $(\Sigma, B)$-unsatisfiable.

To see the ($\Leftarrow$) implication, let $\varphi \in \mathit{QFForm}(\Sigma)$. Without loss of generality we may assume the $\varphi$ is ground (by replacing $\Sigma$ by $\Sigma(Y)$ for $Y$ the variables of $\varphi$ viewed as constants) and a DNF formula $\varphi = \bigvee_{1 \leq i \leq n} (\bigwedge E_i \wedge \bigwedge D_i)$, where each $E_i$ is a finite set of ground $\Sigma$-equations, and each $D_i$ is of the form $D_i = \bigwedge_{u=v \in G_i} u \neq v$ for $G_i$ a finite set of ground equations.

But then $\varphi$ is $(\Sigma, B)$-satisfiable iff $\bigwedge E_i \wedge \bigwedge_{u=v \in G_i} u \neq v$ is $(\Sigma, B)$-satisfiable for some $i$, $1 \leq i \leq n$, iff, by Lemma 4, $E_i \cup B \not\vdash u = v$ for each $u = v \in G_i$. □

**Proof of Theorem** 8

*Proof.* Since the relations $\to^*_{R \cup R(AC_{\Delta})}$ (resp. $\to^*_{R \cup R(AC_{\widehat{\Delta}})}$) just interleave steps of $R$-rewriting with $AC_{\Delta}$-equality (resp. $AC_{\widehat{\Delta}}$-equality) steps, they are commonly denoted, more helpfully and at a higher level, as: $\to^*_{R/AC_{\Delta}}$ (resp. $\to^*_{R/AC_{\widehat{\Delta}}}$), where, by definition, $\to_{R/AC_{\Delta}} = (=_{AC_{\Delta}}) \circ \to_R \circ (=_{AC_{\Delta}})$, and $\to_{R/AC_{\widehat{\Delta}}} = (=_{AC_{\widehat{\Delta}}}$

$)\circ \rightarrow_R \circ(=_{AC_{\widehat{\Delta}}})$. Therefore, they define corresponding binary relations (denoted the same way) on $T_{\Sigma^\square/AC_\Delta}(X)$, resp. $T_{\widehat{\Sigma}/AC_{\widehat{\Delta}}}(X)$, by means of the equivalences: $[u] \rightarrow_{R/AC_\Delta} [v] \;\Leftrightarrow\; (\exists u', v')\ [u] \ni u' \rightarrow_R v' \in [v]$, resp. $[u] \rightarrow_{R/AC_{\widehat{\Delta}}} [v] \;\Leftrightarrow\; (\exists u', v')\ [u] \ni u' \rightarrow_R v' \in [v]$, where $[u], [v]$ abbreviate $AC_\Delta$-equivalence (resp. $AC_{\widehat{\Delta}}$-equivalence) classes. Note, furthermore, that by the assumption that each $g \in \Delta$ has a biggest possible typing with a sort $s_g$ and that that equations $AC_\Delta$ are sort-preserving, reasoning as in the proof of Lemma 3 it is easy to show that for any $u \in T_\Sigma(X)$ its $AC_\Delta$-equivalence class and its $AC_{\widehat{\Delta}}$-equivalence class coincide, so that using $[u]$ for both is unambiguous. Furthermore, since any $t \notin T_\Sigma(X)$ can only have a sort of the form $\top_{[s]}$ for some $[s] \in \widehat{S}$, this also shows that the equations $AC_{\widehat{\Delta}}$ are *sort-preserving* for all terms in $T_\Sigma^\square(X)$. Note, also, that for some $t \notin T_\Sigma(X)$ we may have a strict containment $[t]_{AC_\Delta} \subset [t]_{AC_{\widehat{\Delta}}}$, as the example $a + b \neq_{AC_+} b + a$ in the Introduction shows.

In what follows I summarize some basic facts, terminology, and notation about the relations $\rightarrow_{R/AC_\Delta}$ and $\rightarrow^*_{R/AC_{\widehat{\Delta}}}$. Since all remarks apply to both cases, I will use $\rightarrow^*_{R/AC_\Delta}$ throughout. If $+ \in \Delta$, call a term $u$ a $+$-*term* iff it has the form $u = v + w$, and $+$-*alien term* otherwise. Then the $AC_\Delta$-equivalence class of a $+$-term $u$ is of the form $[q_1 + \ldots + q_n]$, $n \geq 2$, with the $q_1, \ldots, q_n$ $+$-alien subterms of $u$, where, thanks to the associative-commutative nature of $+$, we can *completely disregard both parentheses and the order* among the $q_1, \ldots, q_n$. That is, $[q_1 + \ldots + q_n]$ is a *multiset* whose *elements* are the equivalence classes $[q_1], \ldots, [q_n]$. Therefore, the rewrite relation $[u] \rightarrow_{R/AC_\Delta} [v]$ should be understood as a *multiset-rewriting* relation, but with the proviso that $\Delta$ may have more than one multiset constructor, for example, $+, * \in \Delta$, and rules in $R$ may change such constructors. For example we may have rules like $u + v \rightarrow u' * v'$, where $u' * v'$ is a $+$-alien term, but $*$ is another multiset union operator.

Note that if we have rewrites $[u_1] \rightarrow_{R/AC_\Delta} [v_1]$ and $[u_2] \rightarrow_{R/AC_\Delta} [v_2]$, and $u_1 + u_2 \in T_{\Sigma^\square}(X)$, $v_1 + v_2 \in T_{\Sigma^\square}(X)$, then we also have a *parallel composition* rewrite $[u_1 + u_2] \rightarrow_{R/AC_\Delta} [v_1 + v_2]$ decomposable as, e.g., the sequential composition $[u_1 + u_2] \rightarrow_{R/AC_\Delta} [v_1 + u_2] \rightarrow_{R/AC_\Delta} [v_1 + v_2]$.

Call a rewrite $[u] \rightarrow_{R/AC_\Delta} [v]$ with rule $l \rightarrow r$ a *rewrite at the top* iff there is a $u' \in [u]$ and a substitution $\sigma$ such that $u' = l\sigma$ and $r\sigma \in [v]$; otherwise call $[u] \rightarrow_{R/AC_\Delta} [v]$ a *rewrite below the top*. Furthermore, if $[u] \rightarrow_{R/AC_\Delta} [v]$ is a rewrite below the top with rule $l \rightarrow r$ and substitution $\sigma$, and $u$ is a $+$-term decomposable as $[u] = [q_1 + \ldots + q_n]$, $n \geq 2$, with the $q_i$ $+$-alien subterms, the rewrite $[u] \ni u'[l\sigma]_p \rightarrow_R u'[r\sigma]_p \in [v]$ must satisfy either: (i) $[u'|_p] = [q_{i_1} + \ldots + q_{i_r}]$, $1 \leq i_1 < \ldots < i_r \leq n$, $n > r \geq 2$, so that: (i).1 if $r = n - 1$ then $[u] = [q_j + (l\sigma)]$ and $[v] = [q_j + (r\sigma)]$ for $j$ the only index different from $i_1 < \ldots < i_r$, or (i).2 if $r < n - 1$, then $[u] = [u'' + (l\sigma)]$ and $[v] = [u'' + (r\sigma)]$, with $u''$ the sum of all $q_j$ with $j$ different from $i_1 < \ldots < i_r$; or (ii) $p = p_1 \cdot p_2$ and $[u'|_{p_1}] = [q_i]$ for some $1 \leq i \leq n$, so that $[u] = [q_1 + \ldots + q_{i-1} + u'|_{p_1}[l\sigma]_{p_2} + q_{i+1} + \ldots + q_n]$ and $[v] = [q_1 + \ldots + q_{i-1} + u'|_{p_1}[r\sigma]_{p_2} + q_{i+1} + \ldots + q_n]$. That is, the rewrite either happens modulo $AC_\Delta$ at or below one of the $q_i$, or must rewrite modulo $AC_\Delta$ several, but not all, of the $q_i$.

Using the relations $\to^*_{R/AC_\Delta}$ and $\to^*_{R/AC_{\widehat{\Delta}}}$ we can rephrase the statement of the theorem as the equivalence:

$$[u] \to^*_{R/AC_\Delta} [v] \;\;\Leftrightarrow\;\; [u] \to^*_{R/AC_{\widehat{\Delta}}} [v]$$

for $u, v \in T_\Sigma(X)$, which is a *crucial* requirement, since the example $a + b \neq_{AC_+} b+a$ shows that the equivalence does not hold in general for $u, v \in T_{\Sigma^\square}(X)$. Since the equations $AC_{\widehat{\Delta}}$ are more general than the equations $AC_\Delta$, the ($\Rightarrow$) implication is obvious. To see the ($\Leftarrow$) implication we reason by contradiction and assume that the set $\{([u], n) \in T_{\Sigma/AC_\Delta}(X) \times \mathbb{N} \mid (\exists v \in T_{\Sigma/AC_\Delta}(X))\; [u] \to^n_{R/AC_{\widehat{\Delta}}} [v] \;\wedge\; [u] \not\to^*_{R/AC_\Delta} [v]\}$ is non-empty. Since the term size $|t|$, i.e., the number of nodes of $t$ as a tree, is the same for all terms in an $AC$-equivalence class, we can then give a well-founded lexicographic order to this set by defining $([u], n) > ([u'], m) \;\Leftrightarrow\; |u| > |u'| \vee (|u| = |u'| \wedge n > m)$. Pick a minimal element $([u], n)$ under this order, so that $[u] \to^n_{R/AC_{\widehat{\Delta}}} [v]$ but $[u] \not\to^*_{R/AC_\Delta} [v]$. Let

$$[u] \to_{R/AC_{\widehat{\Delta}}} [w_1] \to_{R/AC_{\widehat{\Delta}}} [w_2] \ldots [w_{n-1}] \to_{R/AC_{\widehat{\Delta}}} [v]$$

be any sequence of the form $[u] \to^n_{R/AC_{\widehat{\Delta}}} [v]$. Let us analyze it carefully. First of all, we must have $w_i \notin T_{\Sigma/AC_\Delta}(X)$, $1 \leq i \leq n-1$, since otherwise $([u], n)$ would not be minimal. Note also that, since for any $u \in T_\Sigma(X)$ its $AC_\Delta$-equivalence class and its $AC_{\widehat{\Delta}}$-equivalence class coincide, $[u] \to_{R/AC_{\widehat{\Delta}}} [v]$ implies $[u] \to_{R/AC_\Delta} [v]$, so we must have $n \geq 2$. Furthermore, $w_i \notin T_{\Sigma/AC_\Delta}(X)$, $1 \leq i \leq n-1$, also means that, since $R$ is a set of $\Sigma$-rules, all $R$-rewrite steps in the sequence for given representatives must happen *below the top*. This rules out the possibility of $u = f(u_1, \ldots, u_k)$ with $f \in (\Sigma - \Delta)$, since this would force $v = f(v_1, \ldots, v_k)$ and rewrites $[u_i] \to^*_{R/AC_{\widehat{\Delta}}} [v_i]$ which, since $|u_i| < |u|$, must also have $[u_i] \to^*_{R/AC_\Delta} [v_i]$, violating $[u] \not\to^*_{R/AC_\Delta} [v]$. Therefore, there must be a symbol in $\Delta$, say, $+$, such that $[u]$ is of the form $[u] = [q_1 + \ldots + q_k]$, $k \geq 2$ with $q_1, \ldots, q_k$ $+$-alien subterms. But then, since all rewrites must happen below the top, the $w_i$, $1 \leq i \leq n-1$ and $[v]$ must all be $+$-terms. Let us now look at the last rewrite $[w_{n-1}] \to_{R/AC_{\widehat{\Delta}}} [v]$. Let $[w_{n-1}] = [q_1 + \ldots + q_l]$ be a decomposition into $+$-alien subterms. It is not only impossible that the rewrite $[w_{n-1}] \to_{R/AC_{\widehat{\Delta}}} [v]$ happened at the top; it is also impossible that it uses a rule in $R$ of the form $w + w' \to r$ with a substitution $\sigma$ such that $(w + w')\sigma =_{AC_{\widehat{\Delta}}} q_{i_1} + \ldots + q_{i_p}$, $1 \leq i_1 < \ldots < i_p \leq l$. This is because then we would have $[w_{n-1}] = [q_{i_1} + \ldots + q_{i_p} + w]$ and $[v] = [r\sigma + w]$, and since $v \in T_\Sigma(X)_{s_+}$, this would force $r\sigma, w, (q_{i_1} + \ldots + q_{i_p}) \in T_\Sigma(X)_{s_+}$ and therefore $w_{n-1} \in T_\Sigma(X)_{s_+}$. Therefore, the rewrite $[w_{n-1}] \to_{R/AC_{\widehat{\Delta}}} [v]$ must happen in one of the $+$-alien subterms of $[w_{n-1}]$, say $q_1$, so that we have $[q_1] \to_{R/AC_{\widehat{\Delta}}} [w']$, and $[v] = [w + w']$. That is, the rewrite $[w_{n-1}] \to_{R/AC_{\widehat{\Delta}}} [v]$ must be of the form $[q_1 + w] \to_{R/AC_{\widehat{\Delta}}} [w + w']$ and, furthermore, we must have $q_1 \notin T_\Sigma(X)_{s_+}$, since otherwise we would have $w_{n-1} \in T_\Sigma(X)_{s_+}$. We now need a lemma:

**Lemma 5.** *(Decomposition Lemma) Let*

$$[u] \to^n_{R/AC_{\widehat{\Delta}}} [q + w],$$

*be a rewrite sequence with* $n \geq 1$*,* $u \in T_\Sigma(X)_{s_+}$*,* $q$ *a* $+$*-alien subterm,* $q \notin T_\Sigma(X)_{s_+}$*, and therefore* $q + w \notin T_\Sigma(X)$*. Then we either have rewrite sequences* $[u] \to^i_{R/AC_{\widehat{\Delta}}} [v] \to^j_{R/AC_{\widehat{\Delta}}} [q{+}w]$ *with* $v \in T_\Sigma(X)_{s_+}$ *a* $+$*-term,* $i{+}j = n$*,* $i, j \geq 1$*, or have a decomposition* $[u] = [u_1 + u_2]$*, where* $u_1, u_2$ *need not be* $+$*-alien, and rewrite sequences:*

*1.* $[u_1] \to^i_{R/AC_{\widehat{\Delta}}} [q]$
*2.* $[u_2] \to^j_{R/AC_{\widehat{\Delta}}} [w]$.

*with* $i + j = n$*,* $i \geq 1$*.*

*Proof.* We reason by strong induction on $n$. **Base Case:** $n = 1$, so that we have $[u] \to_{R/AC_{\widehat{\Delta}}} [q + w]$, say with a rule $l \to r$ and substitution $\sigma$. Equivalently, we have an inverse rewrite $[q + w] \to_{R/AC_{\widehat{\Delta}}} [u]$ with rule $r \to l$ and substitution $\sigma$. Since $q + w \notin T_\Sigma(X)$, the equations $AC_{\widehat{\Delta}}$ are sort-preserving, and $r \in T_\Sigma(X)$, the inverse rewrite $[q{+}w] \to_{R/AC_{\widehat{\Delta}}} [u]$ must happen *below the top* and therefore $u$ must be a $+$-term. Let $q = q_1$ and $[w] = [q_2 + \ldots + q_n]$ with the $q_i$ $+$-alien subterms and $n \geq 2$. (i.e., $w$ could be just $q_2$). That is, the inverse rewrite $[q_1 + \ldots + q_n]] \ni w' \to_R w'[l\sigma]_p \in [v]$, with $w' = w'[r\sigma]_p$, must satisfy either: (i) $[w'|_p] = [q_{i_1} + \ldots + q_{i_l}]$, $1 \leq i_1 < \ldots < i_l \leq n$, $n > l \geq 2$, and either (i).1 if $l = n - 1$ then $[u] = [q_j + (l\sigma)]$ and $[q + w] = [q_j + (r\sigma)]$ for $j$ the only index different from the $i_1, \ldots, i_l$, which is impossible, since by $AC_{\widehat{\Delta}}$-equivalence being sort-preserving and $r \in T_\Sigma(X)$, this would force $j = 1$ and would make $q_1$ $AC_{\widehat{\Delta}}$-equivalent to a $+$-alien subterm of $[u]$, which, again by $AC_{\widehat{\Delta}}$-equivalence being sort-preserving, is impossible since $u$ is a $+$-term, $u \in T_\Sigma(X)_{s_+}$, and $q_1 \notin T_\Sigma(X)_{s_+}$; or (i).2 if $l < n - 1$, then $[u] = [w'' + (l\sigma)]$ and $[q + w] = [w'' + (r\sigma)]$, with $w''$ the sum of all $q_j$ with $j$ different from the $i_1, \ldots, i_l$, which is again impossible for the same reason: $q$ would be an alien subterm of $w''$ and therefore of $u$; or (ii) $p = p_1 \cdot p_2$ and $[w'|_{p_1}] = [q_i]$ for some $1 \leq i \leq n$, so that $[u] = [q_1 + \ldots + q_{i-1} + w'_{p_1}[l\sigma]_{p_2} + q_{i+1} + \ldots + q_n]$ and $[q + w] = [q_1 + \ldots + q_{i-1} + w'_{p_1}[r\sigma]_{p_2} + q_{i+1} + \ldots + q_n]$, which for the same reasons as above forces $i = 1$, giving us the desired decomposition $[u] = [w'_{p_1}[l\sigma]_{p_2} + w]$ splitting the direct rewrite $[u] \to_{R/AC_{\widehat{\Delta}}} [q + w]$ as $[w'_{p_1}[l\sigma]_{p_2}] \to^1_{R/AC_{\widehat{\Delta}}} [q]$ and $[w] \to^0_{R/AC_{\widehat{\Delta}}} [w]$, with $1 + 0 = 1$.

**Induction Step:** Suppose the result holds for any $1 \leq k \leq n$ and consider a sequence of lenght $n + 1$ of the form: $[u] \to^n_{R/AC_{\widehat{\Delta}}} [v] \to_{R/AC_{\widehat{\Delta}}} [q + w]$. Focus on the last rewrite step $[v] \to_{R/AC_{\widehat{\Delta}}} [q + w]$, say with rule $l \to r$ in $R$ and substitution $\sigma$ or, equivalently, on the inverse rewrite $[q + w] \to_{R/AC_{\widehat{\Delta}}} [u]$ with rule $r \to l$ and substitution $\sigma$. As in the base case, this inverse rewrite must happen below the top and $v$ must be a $+$-term. If $v \in T_\Sigma(X)_{s_+}$ we are done. So we may assume $v \notin T_\Sigma(X)_{s_+}$. Let $q = q_1$ and $[w] = [q_2 + \ldots + q_n]$ with the $q_i$ $+$-alien subterms and $n \geq 2$ (i.e., $w$ could be just $q_2$). That is, the inverse rewrite $[q_1 + \ldots + q_n] \ni w' \to_R w'[l\sigma]_p \in [v]$, with $w' = w'[r\sigma]_p$, must satisfy either: (i) $[w'|_p] = [q_{i_1} + \ldots + q_{i_l}]$, $1 \leq i_1 < \ldots < i_l \leq n$, $n > l \geq 2$, which by $q \notin T_\Sigma(X)_{s_+}$ and sort-preservation in equivalence classes forces $q \neq q_{i_1}, \ldots, q_{i_l}$

so that we have either: (i).1 $[v] = [q + (l\sigma)]$ and $[q + w] = [q + (r\sigma)]$, or (i).2 $[v] = [q + w''' + (l\sigma)]$ and $[q + w] = [q + w''' + (r\sigma)]$. In both cases we are done, because $[v] = [q + v']$, so that the induction hypothesis applies to the $n$-step rewrite $[u] \to^n_{R/AC_{\widehat{\Delta}}} [v]$, which either factors through a $[v'']$ with $v'' \in T_\Sigma(X)_{s_+}$ a $+$-term, so that we are done, or splits into $[u_1] \to^i_{R/AC_{\widehat{\Delta}}} [q]$ and $[u_2] \to^j_{R/AC_{\widehat{\Delta}}} [v']$, $i + j = n$, which can each be sequentially composed with the each of the rewrites $[q] \to^0_{R/AC_{\widehat{\Delta}}} [q]$ and $[v'] \to_{R/AC_{\widehat{\Delta}}} [w]$ into which $[v] \to_{R/AC_{\widehat{\Delta}}} [q + w]$ splits to give us the desired decomposition. Otherwise we must have case (ii) with $[w'|_{p_1}] = [q_i]$ and $[v] = [q_1 + \ldots + q_{i-1} + w'_{p_1}[l\sigma]_{p_2} + q_{i+1} + \ldots + q_n]$ and $[q+w] = [q_1 + \ldots + q_{i-1} + w'_{p_1}[r\sigma]_{p_2} + q_{i+1} + \ldots + q_n]$ and we have two possibilities: either $q_i \neq q$, so that we are done by reasoning exactly as in case (i), or $q_i = q$, so that the direct one-step rewrite $[v] \to_{R/AC_{\widehat{\Delta}}} [q + w]$ splits as the parallel composition of $[w'_{p_1}[l\sigma]_{p_2}] \to_{R/AC_{\widehat{\Delta}}} [q]$ and $[w] \to^0_{R/AC_{\widehat{\Delta}}} [w]$.

Since $v \notin T_\Sigma(X)_{s_+}$, $v$ must have a $+$-alien subterm $q'$ with $q' \notin T_\Sigma(X)_{s_+}$, which is either: (1) a $+$-alien subterm of $w'[l\sigma]_{p_2}$, or (2) a $+$-alien subterm of $w$. In case (1), since $[w'|_{p_1}] = [w'|_{p_1}[r\sigma]_{p_2}] = [q]$, if $p_2$ is the empty string, $l$ must be a $+$-alien term, so that $q' = l\sigma$, since the case $l = l_1 + l_2$ is ruled out by $R$ being a set of $\Sigma$-rules, since then $(l_1 + l_2)\sigma \in T_\Sigma(X)_{s_+}$ cannot have $q' \notin T_\Sigma(X)_{s_+}$ as a $+$-alien subterm. But if $p_2$ is non-empty, since $[w'|_{p_1}[r\sigma]_{p_2}] = [q]$, $[w'|_{p_1}[l\sigma]_{p_2}]$ must be a $+$-alien subterm with same top function symbol as $q$, so that $q' = w'|_{p_1}[l\sigma]_{p_2}$. In either case we have $[v] = [q' + w]$ with $q' \notin T_\Sigma(X)_{s_+}$ a $+$-alien subterm, and $[q'] \to_{R/AC_{\widehat{\Delta}}} [q]$, so that the induction hypothesis applies to $[u] \to^n_{R/AC_{\widehat{\Delta}}} [v]$, which either factors through a $+$-term $v' \in T_\Sigma(X)_{s_+}$ and we are done, or splits as the parallel sum of $[u_1] \to^i_{R/AC_{\widehat{\Delta}}} [q']$ and $[u_2] \to^j_{R/AC_{\widehat{\Delta}}} [w]$, with $i + j = n$, giving us the desired splitting of $[u] \to^n_{R/AC_{\widehat{\Delta}}} [v] \to_{R/AC_{\widehat{\Delta}}} [q + w]$ as the parallel composition of $[u_1] \to^i_{R/AC_{\widehat{\Delta}}} [q'] \to_{R/AC_{\widehat{\Delta}}} [q]$ and $[u_2] \to^j_{R/AC_{\widehat{\Delta}}} [w]$.

In case (2) there are two possibilities: (2.1) $[q'] = [q_2]$, $w = [q_2]$, $[v] = [q_2 + w'_{p_1}[l\sigma]_{p_2}]$, and $[q + q_2] = [w'_{p_1}[r\sigma]_{p_2} + q_2]$, so that the induction hypothesis applies to the $n$-step rewrite $[u] \to^n_{R/AC_{\widehat{\Delta}}} [v]$, which either factors through a $+$-term in $T_\Sigma(X)_{s_+}$ and we are done, or splits as the parallel composition of $[u_1] \to^i_{R/AC_{\widehat{\Delta}}} [q_2]$ and $[u_2] \to^j_{R/AC_{\widehat{\Delta}}} [w'_{p_1}[l\sigma]_{p_2}]$, which gives us the desired split of $[u] \to^n_{R/AC_{\widehat{\Delta}}} [v] \to_{R/AC_{\widehat{\Delta}}} [q + q_2]$ as $[u_1] \to^i_{R/AC_{\widehat{\Delta}}} [q_2]$ and $[u_2] \to^j_{R/AC_{\widehat{\Delta}}} [w'_{p_1}[l\sigma]_{p_2}] \to_{R/AC_{\widehat{\Delta}}} [q]$ with $i + j + 1 = n + 1$, or (2.2) $[q'] = [q_j]$, $j \geq 2$, $w = [q_2 + \ldots + q_n]$, $n > 2$, and $[v] = [w + w'_{p_1}[l\sigma]_{p_2}]$. But then the induction hypothesis applies to the $n$-step rewrite $[u] \to^n_{R/AC_{\widehat{\Delta}}} [v]$, which either factors through a $+$-term in $T_\Sigma(X)_{s_+}$ and we are done, or splits as the parallel composition of $[u_1] \to^i_{R/AC_{\widehat{\Delta}}} [q_j]$ and $[u_2] \to^j_{R/AC_{\widehat{\Delta}}} [w'' + w'[l\sigma]_{p_2}]$, with $i \geq 1$, $i + j = n$ and $w'' = [q_2 + \ldots + q_{j-1} + \ldots + q_{j+1} + \ldots + q_n]$ ($w''$ becomes a single $+$-alient subterm when $n = 3$). If $w'' + w'_{p_1}[l\sigma]_{p_2} \in T_\Sigma(X)_{s_+}$ we are done, since we get the factorization $[u_1 + u_2] \to^j_{R/AC_{\widehat{\Delta}}} [u_1 + w'' + w'_{p_1}[l\sigma]_{p_2}] \to^j_{R/AC_{\widehat{\Delta}}} [w + w'_{p_1}[l\sigma]_{p_2}] \to_{R/AC_{\widehat{\Delta}}} [w + q]$ with $u_1 + w'' + w'_{p_1}[l\sigma]_{p_2} \in\in T_\Sigma(X)_{s_+}$, as desired.

Otherwise, we have a composed rewrite $[u_2] \rightarrow^{j}_{R/AC_{\widehat{\Delta}}} [w'' + w'_{p_1}[l\sigma]_{p_2}] \rightarrow_{R/AC_{\widehat{\Delta}}} [q + w'']$ of length $j + 1 \leq n$ to which the induction hypothesis applies, so that, since $w''+w'_{p_1}[l\sigma]_{p_2} \notin T_{\Sigma}(X)_{s_+}$, it either factors through a $+$-term $v' \in T_{\Sigma}(X)_{s_+}$ as $[u_2] \rightarrow^{j.1}_{R/AC_{\widehat{\Delta}}} [v'] \rightarrow^{j.2}_{R/AC_{\widehat{\Delta}}} [w'' + w'_{p_1}[l\sigma]_{p_2}] \rightarrow_{R/AC_{\widehat{\Delta}}} [q + w'']$ with $j.1 + j.2 = j$, and we are done, since then the rewrite $[u] \rightarrow^{n}_{R/AC_{\widehat{\Delta}}} [v] \rightarrow_{R/AC_{\widehat{\Delta}}} [q + w]$ also factors as $[u] \rightarrow^{j.1}_{R/AC_{\widehat{\Delta}}} [u_1 + v] \rightarrow^{j.2}_{R/AC_{\widehat{\Delta}}} [u_1 + w'' + w'_{p_1}[l\sigma]_{p_2}] \rightarrow_{R/AC_{\widehat{\Delta}}} [q + w]$ with $u_1 + v' \in T_{\Sigma}(X)_{s_+}$, as desired, or $[u_2] \rightarrow^{j+1}_{R/AC_{\widehat{\Delta}}} [q + w'']$ splits as the parallel composition of $[u_{2.1}] \rightarrow^{i'}_{R/AC_{\widehat{\Delta}}} [q]$ and $[u_{2.2}] \rightarrow^{j'}_{R/AC_{\widehat{\Delta}}} [w'']$, with $i' + j' = j + 1$, so that, composing $[u_{2.2}] \rightarrow^{j'}_{R/AC_{\widehat{\Delta}}} [w'']$ and $[u_1] \rightarrow^{i}_{R/AC_{\widehat{\Delta}}} [q_j]$ in parallel we get our desired splitting of $[u] \rightarrow^{n}_{R/AC_{\widehat{\Delta}}} [v] \rightarrow_{R/AC_{\widehat{\Delta}}} [q + w]$ as $[u_{2.1}] \rightarrow^{i'}_{R/AC_{\widehat{\Delta}}} [q]$ and $[u_{2.2} + u_1] \rightarrow^{i+j'}_{R/AC_{\widehat{\Delta}}} [q_j + w'']$, with $[w] = [q_j + w'']$, $i' + j' + i = i + j + 1 = n + 1$. This exhausts all cases and finishes the proof of the lemma. $\square$

After this long detour we can finish the proof of Theorem 8. Recall that we had a minimal sequence $[u] \rightarrow^{n}_{R/AC_{\widehat{\Delta}}} [v]$ under the lexicographic order based on pairs $(|u|, n)$ such that $[u] \not\rightarrow^{*}_{R/AC_{\Delta}} [v]$, $n \geq 2$, and $[u] \rightarrow^{n}_{R/AC_{\widehat{\Delta}}} [v]$ factored as $[u] \rightarrow^{n-1}_{R/AC_{\widehat{\Delta}}} [q_1 + w] \rightarrow_{R/AC_{\widehat{\Delta}}} [v]$, with $[v] = [w + w']$, $q_1$ a $+$-alien subterm such that $q_1 \notin T_{\Sigma}(X)_{s_+}$, and $[q_1] \rightarrow_{R/AC_{\widehat{\Delta}}} [w']$. We can then apply the Decomposition Lemma 5 to the sequence $[u] \rightarrow^{n-1}_{R/AC_{\widehat{\Delta}}} [q_1 + w]$ to get a contradiction. If it factors as $[u] \rightarrow^{i}_{R/AC_{\widehat{\Delta}}} [v'] \rightarrow^{j}_{R/AC_{\widehat{\Delta}}} [q_1 + w]$ with $i + j = n - 1$ and $v' \in T_{\Sigma}(X)$, we get a contradiction, because then $[u] \rightarrow^{n}_{R/AC_{\widehat{\Delta}}} [v]$ factors as $[u] \rightarrow^{i}_{R/AC_{\widehat{\Delta}}} [v'] \rightarrow^{j}_{R/AC_{\widehat{\Delta}}} [q_1 + w] \rightarrow_{R/AC_{\widehat{\Delta}}} [v]$, which we have already seen is impossible by the minimality of $[u] \rightarrow^{n}_{R/AC_{\widehat{\Delta}}} [v]$. And if $[u] \rightarrow^{n-1}_{R/AC_{\widehat{\Delta}}} [q_1 + w]$ splits as $[u_1] \rightarrow^{i}_{R/AC_{\widehat{\Delta}}} [q_1]$ and $[u_2] \rightarrow^{j}_{R/AC_{\widehat{\Delta}}} [w]$, $i + j = n - 1$, we get another contradiction, because then $[u] \rightarrow^{n}_{R/AC_{\widehat{\Delta}}} [v]$ splits as $[u_1] \rightarrow^{i+1}_{R/AC_{\widehat{\Delta}}} [w']$ and $[u_2] \rightarrow^{j}_{R/AC_{\widehat{\Delta}}} [w]$, which is impossible since, by the minimality of $[u] \rightarrow^{n}_{R/AC_{\widehat{\Delta}}} [v]$, we must have $[u_1] \rightarrow^{n+i}_{R/AC_{\Delta}} [w']$ and $[u_2] \rightarrow^{n}_{R/AC_{\Delta}} [w]$, whose parallel composition $[u] \rightarrow^{n}_{R/AC_{\Delta}} [v]$ violates the assumption $[u] \not\rightarrow^{*}_{R/AC_{\Delta}} [v]$. $\square$