Combinable Extensions of Abelian Groups

Enrica Nicolini, Christophe Ringeissen, Michael Rusinowitch

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Combinable Extensions of Abelian Groups

Enrica Nicolini — Christophe Ringeissen — Michaël Rusinowitch

N° 6920

Mai 2009

Thème SYM

Rapport de recherche

ISSN 0249-6399 ISRN INRIA/RR--6920--FR+ENG

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

INRIA

centre de recherche NANCY - GRAND EST

Combinable Extensions of Abelian Groups

Enrica Nicolini*, Christophe Ringeissen* , Michaël Rusinowitch*

Thème SYM — Systèmes symboliques
Équipe-Projet Cassis

Rapport de recherche n° 6920 — Mai 2009 — 30 pages

Abstract: The design of decision procedures for combinations of theories sharing some arithmetic fragment is a challenging problem in verification. One possible solution is to apply a combination method à la Nelson-Oppen, like the one developed by Ghilardi for unions of non-disjoint theories. We show how to apply this non-disjoint combination method with the theory of abelian groups as shared theory. We consider the completeness and the effectiveness of this non-disjoint combination method. For the completeness, we show that the theory of abelian groups can be embedded into a theory admitting quantifier elimination. For achieving effectiveness, we rely on a superposition calculus modulo abelian groups that is shown complete for theories of practical interest in verification.

Key-words: Satisfiability Procedure, Combination, Equational Reasoning, Union of Non-Disjoint Theories, Arithmetic, Abelian Groups

* E-mail: FirstName.LastName@loria.fr

Centre de recherche INRIA Nancy – Grand Est
LORIA, Technopôle de Nancy-Brabois, Campus scientifique,
615, rue du Jardin Botanique, BP 101, 54602 Villers-Lès-Nancy
Téléphone : +33 3 83 59 30 00 — Télécopie : +33 3 83 27 83 19

Extensions combinables des groupes abéliens

Résumé : La conception de procédures de décision pour la combinaison de théories partageant un fragment d'arithmétique est un défi dans le domaine de la vérification. Une solution possible consiste à appliquer une méthode de combinaison à la Nelson-Oppen, comme celle développée par Ghilardi pour l'union de théories non-disjointes. On montre comment appliquer cette méthode de combinaison non-disjointe avec la théorie des groupes abéliens comme théorie partagée. On considère la complétude et l'effectivité de cette méthode. Pour la complétude, on montre que la théorie des groupes abéliens peut se plonger dans une théorie admettant l'élimination des quantificateurs. Pour être effectif, on utilise un calcul de superposition modulo la théorie des groupes abéliens qui est montré complet pour des théories intéressantes en pratique dans le domaine de la vérification.

Mots-clés : procédure de satisfiabilité, combinaison, raisonnement équationnel, mélange de théories non-disjointes, groupes abéliens

1 Introduction

Decision procedures are the basic engines of the verification tools used to check the satisfiability of formulae modulo background theories, which may include axiomatizations of standard data-types such lists, arrays, bit-vectors, etc. Nowadays, there is a growing interest in applying theorem provers to construct decision procedures for theories of interest in verification [2, 1, 8, 4]. The problem of incorporating some reasoning modulo arithmetic properties inside theorem provers is particularly challenging. Many works are concerned with the problem of building-in certain equational axioms, starting from the seminal contributions by Plotkin [21] and by Peterson and Stickel [20]. The case of Associativity-Commutativity has been extensively investigated since it appears in many equational theories, and among them, the theory of abelian groups is a very good candidate as fragment of arithmetic. Recently, the standard superposition calculus [19] has been extended to a superposition calculus modulo the built-in theory of abelian groups [12]. This work paves the way for the application of a superposition calculus modulo a fragment of arithmetic to build decision procedures of practical interest in verification. However, practical problems are often expressed in a combination of theories where the fragment of arithmetic is shared by all the other theories involved. In this case the classical Nelson-Oppen combination method cannot be applied since the theories share some arithmetic operators. An extension of the Nelson-Oppen combination method to the non-disjoint case has been proposed in [11]. This non-disjoint combination framework has been recently applied to the theory of Integer Offsets [18]. In this paper, our aim is to consider a more expressive fragment by studying the case of abelian groups.

The contributions of the paper are twofold. First, we show that abelian groups satisfy all the properties required to prove the completeness, the termination and the effectiveness of the non-disjoint extension of the Nelson-Oppen combination method. To prove the completeness, we show the existence of an extension of the theory of abelian groups having quantifier elimination and that behaves the same w.r.t. the satisfiability of constraints. Second, we identify a class of theories that extend the theory of abelian groups and for which a simplified constraint-free (but many-sorted) version of the superposition calculus introduced in [12] is proved to be complete. This superposition calculus allows us to obtain effective decision procedures that can be plugged into the non-disjoint extension of the Nelson-Oppen combination method.

This paper is organized as follows. Section 2 briefly introduces the main concepts and the non-disjoint combination framework. In Section 3, we show some very useful properties in order to use the theory of abelian groups, namely AG, in the non-disjoint combination framework, especially we prove the quantifier elimination of a theory that is an extension of AG. In Section 4, we present a calculus modulo AG. In Section 5, we show its refutational completeness and we study how this calculus may lead to combinable decision procedures. Examples are given in Section 6. We conclude with some final remarks in Section 7. Most of the proofs are omitted and can be found in the appendix.

2 Preliminaries

We consider a many-sorted language. A *signature* Σ is a set of sorts, functions and predicate symbols (each endowed with the corresponding arity and sort). We assume that, for each sort s, the equality "$=_s$" is a logical constant that does not occur in Σ and that is always interpreted as the identity relation over (the interpretation of) s; moreover, as a notational convention, we will often omit the subscript and we will shorten $=$ and $\neq$ with $\bowtie$. Again, as a matter of convention, we denote with $\Sigma^{\underline{a}}$ the signature obtained from Σ by adding a set $\underline{a}$ of new constants (each of them again equipped with its sort), and with $t\theta$ the application of a substitution θ to a term t. Σ-*atoms*, Σ-*literals*, Σ-*clauses*, and Σ-*formulae* are defined in the usual way, i.e. they must respect the arities of function and predicate symbols and the variables occurring in them must also be equipped with sorts (well-sortedness). The empty clause is denoted by $\Box$. A set of Σ-literals is called a Σ-*constraint*. Terms, literals, clauses and formulae are called *ground* whenever no variable appears in them; *sentences* are formulae in which free variables do not occur.

From the semantic side, we have the standard notion of a Σ-*structure* $\mathcal{M}$: it consists of non-empty pairwise disjoint domains M_s for every sort s and a sort- and arity-matching interpretation $\mathcal{I}$ of the function and predicate symbols from Σ. The truth of a Σ-formula in $\mathcal{M}$ is defined in any one of the standard ways. If $\Sigma_0 \subseteq \Sigma$ is a subsignature of Σ and if $\mathcal{M}$ is a Σ-structure, the Σ_0-*reduct* of $\mathcal{M}$ is the Σ_0-structure $\mathcal{M}_{|\Sigma_0}$ obtained from $\mathcal{M}$ by forgetting the interpretation of the symbols from $\Sigma \setminus \Sigma_0$.

A collection of Σ-sentences is a Σ-theory, and a Σ-theory T admits (or has) *quantifier elimination* iff for every formula $\varphi(\underline{x})$ there is a quantifier-free formula (over the same free variables $\underline{x}$) $\varphi'(\underline{x})$ such that $T \models \varphi(\underline{x}) \leftrightarrow \varphi'(\underline{x})$.

In this paper, we are concerned with the *(constraint) satisfiability problem* for a theory T, which is the problem of deciding whether a Σ-constraint is satisfiable in a model of T. Notice that a constraint may contain variables: since these variables may be equivalently replaced by free constants, we can reformulate the constraint satisfiability problem as the problem of deciding whether a finite conjunction of ground literals in a simply expanded signature $\Sigma^{\underline{a}}$ is true in a $\Sigma^{\underline{a}}$-structure whose Σ-reduct is a model of T.

2.1 A Brief Overview on Non-Disjoint Combination

Let us consider now the constraint satisfiability problem w.r.t. a theory T that is the union of the two theories $T_1 \cup T_2$, and let us suppose that we have at our disposal two decision procedures for the constraint satisfiability problem w.r.t. T_1 and T_2 respectively. It is known (cf. [5]) that such a problem without any other assumption on T_1 and T_2 is undecidable; nevertheless, the following theorem holds:

Theorem 1 ([11]) *Consider two theories T_1, T_2 in signatures Σ_1, Σ_2 such that:*

1. *both T_1, T_2 have a decidable constraint satisfiability problem;*
2. *there is some universal theory T_0 in the signature $\Sigma_0 := \Sigma_1 \cap \Sigma_2$ such that:*
   (a) *T_1, T_2 are both T_0-compatible;*

(b) T_1, T_2 *are both effectively Noetherian extensions of* T_0.

Then the $(\Sigma_1 \cup \Sigma_2)$*-theory* $T_1 \cup T_2$ *also has a decidable constraint satisfiability problem.*

The procedure underlying Theorem 1 basically extends the Nelson-Oppen combination method [17] to theories over non disjoint signatures, thus lifting the decidability of the constraint satisfiability problem from the component theories to their union.

The requirement (2a) of T_0-*compatibility* over the theories T_1 and T_2 means that there is a Σ_0-theory $T_0^\star$ such that (i) $T_0 \subseteq T_0^\star$; (ii) $T_0^\star$ has quantifier elimination; (iii) every Σ_0-constraint which is satisfiable in a model of T_0 is satisfiable also in a model of $T_0^\star$; and (iv) every Σ_i-constraint which is satisfiable in a model of T_i is satisfiable also in a model of $T_0^\star \cup T_i$, for $i = 1, 2$. This requirement guarantees the completeness of the combination procedure underlying Theorem 1 and generalizes the stable infiniteness requirement used for the completeness of the original Nelson-Oppen combination procedure.

The requirement (2b) on T_1, T_2 of being effectively Noetherian extensions of T_0 means the following: first of all (i) T_0 is *Noetherian*, i.e., for every *finite* set of free constants $\underline{a}$, every infinite ascending chain $\Theta_1 \subseteq \Theta_2 \subseteq \cdots \subseteq \Theta_n \subseteq \cdots$ of sets of ground $\Sigma_0^{\underline{a}}$-atoms is eventually constant modulo T_0, i.e. there is a Θ_n in the chain such that $T_0 \cup \Theta_n \models \Theta_m$, for every natural number m. Moreover, we require to be capable to (ii) compute T_0-*bases* for both T_1 and T_2, meaning that, given a finite set Γ_i of ground clauses (built out of symbols from Σ_i and possibly further free constants) and a finite set of free constants $\underline{a}$, we can always compute a finite set Δ_i of *positive* ground $\Sigma_0^{\underline{a}}$-clauses such that *(a)* $T_i \cup \Gamma_i \models C$, for all $C \in \Delta_i$ and *(b)* if $T_i \cup \Gamma_i \models D$ then $T_0 \cup \Delta_i \models D$, for every positive ground $\Sigma_0^{\underline{a}}$-clause D $(i = 1, 2)$. Note that if Γ_i is T_i-unsatisfiable then w.l.o.g. $\Delta_i = \{\Box\}$. Intuitively, the Noetherianity of T_0 means that, fixed a finite set of constants, there exists only a finite number of atoms that are not redundant when reasoning modulo T_0; on the other hand, the capability of computing T_0-bases means that, for every set Γ_i of ground $\Sigma_i^{\underline{a}}$-literals, it is possible to compute a finite "complete set" of logical consequences of Γ_i over Σ_0; these consequences over the shared signature are exchanged between the satisfiability procedures of T_1 and T_2 in the loop of the combination procedure à la Nelson-Oppen, whose termination is ensured by the Noetherianity of T_0.

We depict in the algorithm below the combination procedure, where Γ_i denotes a set of ground literals built out of symbols of Σ_i (for $i = 1, 2$), a set of shared free constants $\underline{a}$ and possibly further free constants.

Algorithm 1 Extending Nelson-Oppen

1. If $T_0\text{-}\mathsf{basis}_{T_i}(\Gamma_i) = \Delta_i$ and $\Box \notin \Delta_i$ for each $i \in \{1, 2\}$, then
  1.1. For each $D \in \Delta_i$ such that $T_j \cup \Gamma_j \not\models D$, $(i \neq j)$, add D to Γ_j
  1.2. If Γ_1 or Γ_2 has been changed in **1.1**, then rerun **1.**
 Else **return** *"unsatisfiable"*
2. If this step is reached, **return** *"satisfiable"*.

In what follows we see how to apply this combination algorithm in order to show the decidability of the constraint satisfiability problem for the union of theories that share the theory of abelian groups, denoted from now on by

AG. To this aim, we first show that AG is Noetherian (Section 3.2). Second, we exhibit a theory $AG^* \supseteq AG$ that admits quantifier-elimination and whose behaviour w.r.t. the satisfiability of constraints is the same of AG (Section 3.3). Third, we see how to construct effectively Noetherian extensions of AG by using a superposition calculus (Section 5.1).

3 The Theory of Abelian Groups

In this section we focus on some properties that are particularly useful when trying to apply Theorem 1 to a combination of theories sharing AG.

$\boxed{AG}$ rules the behaviour of the binary function symbol $+$, of the unary function symbol $-$ and of the constant 0. More precisely, $\Sigma_{AG} := \{0 : \text{AG}, - : \text{AG} \to \text{AG}, + : \text{AG} \times \text{AG} \to \text{AG}\}$, and AG is axiomatized as follows:

$$\begin{array}{ll} \forall x, y, z \;\; (x + y) + z = x + (y + z) & \qquad \forall x, y \;\; x + y = y + x \\ \forall x \;\; x + 0 = x & \qquad \forall x \;\; x + (-x) = 0 \end{array}$$

From now on, given an expansion of Σ_{AG}, a generic term of sort AG will be written as $n_1 t_1 + \cdots + n_k t_k$, where t_i is a term whose root symbol is different both from $+$ and $-$, $t_1 - t_2$ is a shortening for $t_1 + (-t_2)$, and $n_i t_i$ is a shortening for $t_i + \cdots + t_i$ (n_i)-times if n_i is a positive integer, or $-t_i - \cdots - t_i$ $(-n_i)$-times if n_i is negative.

3.1 Unification in Abelian Groups

We will consider a superposition calculus using unification in AG with free symbols, which is known to be finitary [6]. In the following, we restrict ourselves to particular AG-unification problems with free symbols in which no variables of sort AG occur. By using a straightforward many-sorted extension of the Baader-Schulz combination procedure [3], one can show that an AG-equality checker is sufficient to construct a complete set of unifiers for these particular AG-unification problems with free symbols. Moreover, the following holds:

Lemma 1 *Let Γ be a AG-unification problem with free symbols in which no variable of sort* AG *occurs, and let $CSU_{AG}(\Gamma)$ be a complete set of AG-unifiers of Γ. For any $\mu \in CSU_{AG}(\Gamma)$, we have that* 1.) *$VRan(\mu) \subseteq Var(\Gamma)$, and that,* 2.) *for any AG-unifier σ of Γ such that $Dom(\sigma) = Var(\Gamma)$, there exists $\mu \in CSU_{AG}(\Gamma)$ such that $\sigma =_{AG} \mu(\sigma_{|VRan(\mu)})$.*

3.2 Noetherianity of Abelian Groups

Let us start by proving the Noetherianity of AG; the problem of discovering effective Noetherian extensions of AG will be addressed in Section 5.1, after the introduction of an appropriate superposition calculus (Section 4).

Proposition 1 *AG is Noetherian.*

Proof. Note that any equation is AG-equivalent to $(\sharp)$ $\sum_{i=1}^{k} n_i a_i = \sum_{j=1}^{h} m_j b_j$, where a_i, b_j are free constants in $\underline{a} \cup \underline{b}$ and n_i, m_j are positive integers, so we can restrict ourselves to chains of sets of equations of the kind ($\sharp$). Theorem 3.11 in [7] shows that AC is Noetherian, where AC is the theory of an associative and commutative $+$ (thus $\Sigma_{AC} = \{+\}$). From the definition of Noetherianity it follows that, if T is a Noetherian Σ-theory, any other Σ-theory T' such that $T \subseteq T'$ is Noetherian, too. Clearly, the set of sentences over Σ_{AC} implied by AG extends AC; hence any ascending chain of sets of equations of the kind ($\sharp$) is eventually constant modulo AG, too.

In order to apply Theorem 1 to a combination of theories that share AG, we need to find an extension of AG that admits quantifier elimination and such that any constraint is satisfiable w.r.t. such an extension iff it is already satisfiable w.r.t. AG. A first, natural candidate would be AG itself. Unfortunately it is not the case: more precisely, it is known that AG cannot admit quantifier elimination (Theorem A.1.4 in [14]). On the other hand, it is possible to find an extension AG^* with the required properties: AG^* is the theory of divisible abelian groups with infinitely many elements of each finite order.

3.3 An Extension of Abelian Groups having Quantifier Elimination

Let $D_n := \forall x \exists y\, ny = x$, let $O_n(x) := nx = 0$ and let $L_{m,n} := \exists y_1, y_2, \ldots, y_m \bigwedge_{i \neq j} y_i \neq y_j \wedge \bigwedge_{i=1}^{m} O_n(y_i)$, for $n, m \in \mathbb{N}$. D_n expresses the fact that each element is divisible by n, $O_n(x)$ expresses that the element x is of order n, and $L_{m,n}$ expresses the fact that there exist at least m elements of order n. The theory AG^* of divisible abelian groups with infinitely many elements of each finite order can be thus axiomatized by $AG \cup \{D_n\}_{n>1} \cup \{L_{m,n}\}_{m>0,n>1}$.

Now, instead of showing directly that AG^* admits quantifier elimination and satisfies exactly the same constraints that are satisfiable w.r.t. AG, we rely on a different approach. Let us start by introducing some more notions about structures and their properties. Given a Σ-structure $\mathcal{M} = (M, \mathcal{I})$, let Σ^M be the signature where we add to Σ constant symbols m for each element of M. The *diagram* $\Delta(\mathcal{M})$ of $\mathcal{M}$ is the set of all the ground Σ^M-literals that are true in $\mathcal{M}$. Given two Σ-structures $\mathcal{M} = (M, \mathcal{I})$ and $\mathcal{N} = (N, \mathcal{J})$, a Σ-*embedding* (or, simply, an embedding) between $\mathcal{M}$ and $\mathcal{N}$ is a mapping $\mu : M \to N$ among the corresponding support sets satisfying, for all the Σ^M-atoms ψ, the condition $\mathcal{M} \models \psi$ iff $\mathcal{N} \models \psi$ (here $\mathcal{M}$ is regarded as a Σ^M-structure, by interpreting each additional constant $a \in M$ into itself, and $\mathcal{N}$ is regarded as a Σ^M-structure by interpreting each additional constant $a \in M$ into $\mu(a)$). If $M \subseteq N$ and if the embedding $\mu : M \to N$ is just the identity inclusion $M \subseteq N$, we say that $\mathcal{M}$ is a *substructure* of $\mathcal{N}$. If it happens that, given three models of T: $\mathcal{A}$, $\mathcal{M}$, $\mathcal{N}$ and two embeddings $f : \mathcal{A} \to \mathcal{M}$ and $g : \mathcal{A} \to \mathcal{N}$, there always exists another model of T, $\mathcal{H}$, and two embeddings $h : \mathcal{M} \to \mathcal{H}$ and $k : \mathcal{N} \to \mathcal{H}$ such that the composition $f \circ h = g \circ k$, we say that T has the *amalgamation property.* Finally if, given a Σ-theory T and a model $\mathcal{M}$ for T, it happens that, for each Σ-sentence ψ, $\mathcal{M} \models \psi$ if and only if $T \models \psi$, then we say that T is a *complete* theory.

Now, in [14], Exercise 8 page 380, it is stated that AG^* is the so-called *model companion* of the theory AG, meaning that *(i)* for each model $\mathcal{M}$ of AG^* the theory $AG^* \cup \Delta(\mathcal{M})$ is a complete theory, *(ii)* every constraint that is satisfiable in a model of AG is satisfiable in a model of AG^* and *(iii)* every constraint that is satisfiable in a model of AG^* is satisfiable in a model of AG

(of course, since $AG \subset AG^*$, condition *(iii)* gets trivial, but we report here for sake of completeness). At this point, since the behaviour of AG and AG^* is the same w.r.t. the satisfiability of constraints, the only condition that remains to be verified is that AG^* admits quantifier elimination. But:

Theorem 2 *AG has the amalgamation property.*

Corollary 1 *AG^* admits quantifier elimination.*

Proof.
In [10] it is shown that, if T is a universal theory and T^* is a model-companion of T, then the following are equivalent:

(i) T^* has quantifier elimination;

(ii) T has the amalgamation property.

Since AG has the amalgamation property, and AG^* is the model-companion of AG, we have that AG^* has quantifier elimination.

4 A Calculus for Abelian Groups

In [12] the authors give a superposition calculus in which the reasoning about elements of an abelian group is completely built-in. Our aim is to elaborate that calculus so that it provides a decision procedure for the satisfiability problem modulo theories modelling interesting data structures and extending AG. More precisely, we want to produce a calculus able to check the satisfiability in the models of AG of sets of literals in the form $Ax(T) \cup G$, where $Ax(T)$ is a set of unit clauses, not necessarily ground, formalizing the behaviour of some data structure, and G is a set of ground literals. To that purpose, we eliminate the constraints from the calculus and we use a many-sorted language that extends the signature of the theory of abelian groups Σ_{AG} by additional function symbols. Moreover, we will adopt from now on the following assumption: we will consider only

unit clauses with no occurrence of variables of sort AG. $(*)$

Let us start to see more in detail the notations and the concepts used in the rules of the calculus.

First of all, we will reason over terms modulo an AG-rewriting system: quoting [12], the system R_{AG} consists of the rules (i) $x+0 \to 0$, (ii) $-x+x \to 0$, (iii) $-(-x) \to 0$, (iv) $-0 \to 0$, (v) $-(x+y) \to (-x)+(-y)$. Moreover, rewriting w.r.t. R_{AG} is considered *modulo AC*, namely the associativity and the commutativity of the $+$, thus, when rewriting $\to_{R_{AG}}$, we mean the relation $=_{AC}\!\to_{R_{AG}}\!=_{AC}$. The normal form of a term t w.r.t. R_{AG} will be often written as AG-nf(t), and two terms t_1 and t_2 are equal modulo AG iff AG-nf$(t_1) =_{AC} AG$-nf(t_2). Accordingly, we say that a substitution σ is in AG-normal form whenever all the terms occurring in the codomain of σ are in AG-normal form.

Moreover, we will consider an order $\succ$ over terms that is total, well-founded, strict on ground terms and such that 1. $\succ$ is AC-compatible, meaning that $s' =_{AC} s \succ t =_{AC} t'$ implies $s' \succ t'$, 2. $\succ$ orients all the rules of R_{AG}, meaning that $l\sigma \succ r\sigma$ for every rule $l \to r$ of R_{AG} and all the grounding substitutions σ; 3. $\succ$ is monotonic on ground terms, meaning that for all ground terms s, t, u, $u[s]_p \succ u[t]_p$ whenever $s \succ t$. An ordering satisfying all the requirements above can be easily obtained considering an RPO ordering with a total precedence $\succ_\Sigma$ on the symbols of the signature Σ such that $f \succ_\Sigma - \succ_\Sigma + \succ_\Sigma 0$ for all symbols f in Σ and such that all the symbols have a lexicographic status, except $+$, whose status is multiset (see [9], where, in order to compare two terms, the arity of $+$ is considered variable, but always greater than 1).

As a last convention, with a little abuse of notation, we will call *summand* any term whose root symbol is different from both $+$ and $-$, notwithstanding its sort. In this way a *generic* term can be written in the shape $n_1t_1 + \cdots + n_kt_k$ (if it is of sort different from AG, it simply boils down to t_1).

Now, we are ready to describe the calculus. We will rely basically on three rules, *Direct AG-superposition, Inverse AG-superposition* and *Reflection*, and, as in [12], we will apply the rules only in case the premises satisfy certain conditions as explained in the following. Moreover, from now on we assume that all the literals will be eagerly maintained in *AG*-normal form, meaning that they will be maintained as (dis)equations between terms in *AG*-normal form.

Orientation for the left premises of direct AG-superposition Let $l = r$ be an equation; if it is on the sort AG, then it can be equivalently rewritten into $e = 0$. Thus the term e is a term of the form $n_1t_1 + n_2t_2 + \cdots + n_pt_p$, where the t_i are non variable distinct summands, and the n_i's are non zero integers. By splitting the summands into two disjoint sets, the equation $e = 0$ can be rewritten as $n_1t_1 + \cdots + n_kt_k = -n_{k+1}t_{k+1} - \cdots - n_pt_p$. In the following, we will call any equation over AG in that form an *orientation* for $e = 0$. If $l = r$ is an equation over a sort different from AG, then an *orientation* of $l = r$ will be either $l = r$ or $r = l$.

Orientation for the left premises of inverse AG-superposition Let $e = 0$ be an equation over the sort AG. If e or $-e$ is a term of the form $s + e'$, where s is a summand that occurs positively and e' is a generic term, then $-s = e'$ is an *inverse orientation* for $e = 0$.

Splitting of the right premises for direct AG-superposition Let t be a non-variable subterm of either r or s in the literal $r \bowtie s$; moreover, if s is of sort AG, we can freely assume that s is 0. If t is of sort AG, we ask that t is not immediately under $+$ nor under $-$, and that the root of t is different from $-$. Thus, we can imagine that t is of the kind $n_1s_1 + \cdots + n_ps_p + t'$, where all s_i are distinct summands, all n_i are positive integers and t' contains only negative summands. In this case, $t_1 + t_2$ is a *splitting* for t if t_1 is a term of the form $k_1s_1 + \cdots + k_ps_p$, where $0 \leq k_i \leq n_i$, and t_2 is $(n_1 - k_1)s_1 + \cdots + (n_p - k_p)s_p + t'$. If t is not over the sort AG, then the only splitting admissible for t is t itself.

Splitting of the right premises for inverse AG-superposition Let t be a non variable subterm of either r or s in the literal $r \bowtie s$; moreover, if s is of sort AG, we can freely assume that s is 0. Let t be of sort AG, and let t be not immediately below $+$ nor $-$. If t is of the form $-s+t'$, where s is a summand, then t_1+t_2 is an *inverse splitting* for t if t_1 is $-s$ and t_2 is t'.

AG-superposition rules In the left premise $l=r$ of the direct AG-superposition rule, it is assumed that $l=r$ is an orientation of the literal. Similarly, in the right premise, $D[t_1+t_2]_p$ denotes that $D_{|_p}$ is a non-variable term that is not immediately below $+$ or $-$ with a splitting t_1+t_2. Similarly, in the inverse AG-superposition rule, $l=r$ and $D_{|_p}$ denote inverse orientation and splitting, respectively. The inference system, denoted by $\mathcal{SP}_{AG}$, is made of the following rules:

Direct AG-superposition	$\dfrac{l=r \qquad D[t_1+t_2]_p}{(D[r+t_2]_p)\mu_i}$	(i)
Inverse AG-superposition	$\dfrac{l=r \qquad D[t_1+t_2]_p}{(D[r+t_2]_p)\mu_i}$	(ii)
Reflection	$\dfrac{u' \neq u}{\square}$	(iii)

The condition (i) is that μ_i is a most general solution of the AG-unification problem $l =_{AG} t_1$; moreover the inference has to be performed whenever there is a ground instantiation of μ_i, θ, s.t., if $nu=s$ is the AG-normal form of $(l=r)\mu_i\theta$ and $D'[nu]_q$ is the AG-normal form of $(D[t_1+t_2]_p)\mu_i\theta$ in which, in position q, nu appears as subterm, then (a) $u \succ s$, (b) nu appears as subterm of the maximal term in D'.

The condition (ii) is that μ_i is a most general solution of the AG-unification problem $l =_{AG} t_1$; moreover the inference is needed to be performed whenever there is a ground instantiation of μ_i, θ, s.t., if $-u=s$ is the AG-normal form of $(l=r)\mu_i\theta$ and $D'[-u]_q$ is the AG-normal form of $(D[t_1+t_2]_p)\mu_i\theta$ in which, in position q, $-u$ appears as subterm, then (a) either u is the maximal summand in s or $u \succ s$, (b) $-u$ appears as subterm of the maximal term in D'.

The condition (iii) is that the AG-unification problem $u =_{AG} u'$ has a solution (and $\square$ is the syntactic convention for the empty clause).

Moreover, we assume that, after each inference step, the newly-derived literal is normalized modulo AG.

We point out that, thanks to Lemma 1(1.) and to our assumption $(*)$, at any step of a saturation no variable of sort AG is introduced, thus the resulting saturated set will consist of literals in which no variable of sort AG occurs. Moreover, we can note that the conditions on the inferences are, in general, far from being obvious to check. However, for our purposes, we will often perform inferences involving at least one ground literal. In that case, verifying all the conditions becomes easier.

5 Refutational Completeness of $\mathcal{SP}_{AG}$

In order to prove the refutational completeness of the calculus presented above, we will adapt the model generation technique presented in [12]. The idea behind this technique consists in associating to any saturated set of literals that does not contain the empty clause a model of terms identified modulo a rewriting system, the latter being built according to some of the equations in the saturated set. Even if in our calculus no constrained literal will appear, in order to build the model of terms we will rely only on ground instances of the literals in the saturation that are *irreducible*. Moving from [12] and extending to the many-sorted case, we say that:

Definition 1 *An equation $s = t$ is in* one-sided form *whenever, (a) if s and t are of sort* AG, *the equation is in the form $e = 0$, and e is in AG-normal form; (b) if s and t are not of sort* AG, *both s and t are in AG-normal form.*

Whereas an equation over a sort different from AG has a unique one-sided form, an equation over the sort AG has two *AG*-equivalent one-sided forms, but in what follows it does not matter which of the two will be considered. Thus, from now on, when we will refer to equations, we will always assume that the equations are in one-sided form.

Definition 2 *Let s be a term, σ be a grounding substitution such that both σ and s are in AG-normal form. Moreover, let R be a ground term rewriting system. We will say that the $maxred_R(s\sigma)$ is*

- 0*, if AG-nf$(s\sigma)$ is R-irreducible;*
- $\max PS$*, where PS is the following set of terms (ordered w.r.t. $\succ$):*
  $PS := \{u$ *is a summand* $|$ *for some term v and some n in $\mathbb{Z}$, AG-nf$(s\sigma)$ is of the form $nu + v$ and nu is R-reducible*$\}$.

Definition 3 [1] *Let s be a term in which no variable of sort* AG *occurs, let σ be a grounding substitution such that both s and σ are in AG-normal form, and let R be a ground TRS. The pair (s, σ) is* irreducible *w.r.t. R whenever:*

- *AG-nf$(s\sigma)$ is R-irreducible, or*
- *if AG-nf$(s\sigma)$ is R-reducible, let u be the $maxred_R(s\sigma)$. Then, (s, σ) is irreducible if s is not a variable and, for each term of the form $t = f(t_1, \ldots, t_n)$ such that s is of the form $t+v$ or $-t+v$ or t and such that $u \succeq$ AG-nf$(t\sigma)$, each (t_i, σ) is irreducible.*

If L is a literal, the pair (L, σ) is irreducible w.r.t. R:

- *if L is an (dis)equation whose one-sided form is of the form $e \bowtie 0$, then (e, σ) is irreducible w.r.t. R;*
- *if L is an (dis)equation whose one-sided form is of the form $s \bowtie t$, both (s, σ) and (t, σ) are irreducible w.r.t. R.*

Before going on with the description of all the ingredients that are needed in order to show the completeness of the calculus, we want to point out a property that will be useful in the following.

[1]Here we are adapting, in case of absence of variables of sort AG, the definition of recursive irreducibility of [12], but in our context the two notions of recursive irreducibility and irreducibility are collapsing.

Proposition 2 *Let s be a term in which no variable of sort* AG *occurs, let σ be a grounding substitution such that both s and σ are in AG-normal form, and let R be a ground TRS such that (s, σ) is irreducible w.r.t. R. Moreover, let $\sigma =_{AG} \mu\pi$, where π is another grounding substitution in AG-normal form and μ is a substitution that does not have variables of sort* AG *in its range. Then $(s\mu, \pi)$ is still irreducible w.r.t. R.*

To extract, from a given set of ground literals, a term rewriting system, we first of all transform all the equations in reductive normal form (see [12]):

Definition 4 *A ground literal $s \bowtie t$ in AG-normal form is in* reductive form *whenever s is of the form nu, t is the form $n_1 v_1 + \cdots + n_k v_k$ and $n > 0$, n_i are non-zero integers, u and v_i are summands with $u \succ v_i$.*

Of course, if s and t are of sort different from AG, the definition above simply says that $s \succ t$; moreover, it is always possible, given an equation, to obtain an equivalent one in reductive normal form. Now, a term rewriting system is obtained as follows:

Definition 5 *Let S be a set of literals, let L be an equation with a ground instance $L\sigma$, let G be the reductive form of $L\sigma$: $G \equiv nu = r$. Then G* generates *the rule $nu \to r$ if the following conditions are satisfied:*

(i) $(R_G \cup AG) \not\models G$;

(ii) $u \succ r$;

(iii) nu *is* R_G*-irreducible;*

(iv) (L, σ) *is irreducible w.r.t.* R_G.

where R_G is the set of rules generated by the reductive forms of the ground instances of S that are smaller than G w.r.t. $\succ$. Moreover, if $n > 1$, then also the rule $-u \to (n-1)u - r$ is generated.

Now, exactly as in [12], we associate to a generic set of literals saturated under the rules of our calculus and that does not contain the empty clause, S, a structure I that is an AG-model for S. I is the equality Herbrand interpretation defined as the congruence on ground terms generated by $R_S \cup AG$, where R_S is the set of rules generated by S according to Definition 5. Since we are in a many-sorted context, the domain of I consists of different sets, one for each sort; since the rewriting rules in $R_S \cup AG$ are sort-preserving, the congruence on the ground terms is well-defined. Applying the same kind of arguments used to prove Lemma 10 in [12], we have that $R_S \cup AG$ is terminating and confluent, and it still holds that $I \models s = t$ iff $s \to^*_{R_S \cup R_{AG}} \tau \leftarrow^*_{R_S \cup R_{AG}} t$ for some term τ. To show that I is really an AG-model for S, we can prove the following lemma:

Lemma 2 *Let S be the closure under the calculus of a set of literals S_0, and let us assume that the empty clause does not belong to S. Let I be the model of terms derived from S as described above, and let $Ir_{R_S}(S)$ be the set of ground instances $L\sigma$ of L in S such that (L, σ) is irreducible w.r.t. R_S. Then (1) $I \models Ir_{R_S}(S)$ implies that $I \models S$, and (2) $I \models Ir_{R_S}(S)$.*

From the lemma above, it follows immediately:

Theorem 3 *The calculus $\mathcal{SP}_{AG}$ is refutational complete for any set of literals that do not contain variables of sort* AG.

5.1 Computing AG-bases

Let us go back, for the moment, to Theorem 1, and especially to condition (2b) that states that, in order to apply a combination procedure à la Nelson-Oppen to a pair of theories T_1 and T_2 sharing AG, we have to ensure that T_1 and T_2 are effectively Noetherian extensions of AG, i.e. we have to ensure the capability of computing AG-bases for T_1 and T_2. Let us suppose that T_1 and T_2 are Σ_i-theory whose set of axioms is described by a finite number of unit clauses.

Now, for $i = 1, 2$, let Γ_i be a set of ground literals over an expansion of $\Sigma_i \supseteq \Sigma_{AG}$ with the finite sets of fresh constants $\underline{a}, \underline{b}_i$, and suppose to perform a saturation w.r.t. $\mathcal{SP}_{AG}$ adopting an RPO ordering in which the precedence is $f \succ a \succ - \succ + \succ 0$ for every function symbol f in $\Sigma_i^{\underline{b}_i}$ different from $+, -, 0$, every constant a in $\underline{a}$ and that all the symbols have a lexicographic status, except $+$, whose status is multiset. Relying on the refutational completeness of $\mathcal{SP}_{AG}$, Proposition 3 shows how $\mathcal{SP}_{AG}$ can be used in order to ensure that T_1 and T_2 are effectively Noetherian extensions of AG:

Proposition 3 *Let S be a finite saturation of $T_i \cup \Gamma_i$ w.r.t $\mathcal{SP}_{AG}$ not containing the empty clause and suppose that, in every equation $e = 0$ containing at least one of the constants a in $\underline{a}$ as summand, the maximal summand is not unifiable with any other summand in e. Then the set Δ_i of all the ground equations over $\Sigma_{AG}^{\underline{a}}$ in S is an AG-basis for T_i w.r.t. $\underline{a}$ $(i = 1, 2)$.*

6 Some Examples

Theorem 3 guarantees that $\mathcal{SP}_{AG}$ is refutational complete, thus, if we want to turn it into a decision procedure for the constraint satisfiability problem w.r.t. a theory of the kind $T \cup AG$, it is sufficient to prove that any saturation under the rules of $\mathcal{SP}_{AG}$ of a set of ground literals and the axioms of T is finite. Let us show some examples in which this is actually the case.

Lists with Length The theory of lists with length can be seen as the union of the theories $T_L \cup T_\ell \cup AG$, with T_L being the theory of lists and T_ℓ being the theory that axiomatizes the behaviour of the function for the length; more formally:

$\boxed{T_L}$ has the many-sorted signature of the theory of lists: Σ_L is the set of function symbols $\{\mathsf{nil} : \text{LISTS}, \mathsf{car} : \text{LISTS} \to \text{ELEM}, \mathsf{cdr} : \text{LISTS} \to \text{LISTS}, \mathsf{cons} : \text{ELEM} \times \text{LISTS} \to \text{LISTS}\}$ plus the predicate symbol $\mathsf{atom} : \text{LISTS}$, and it is axiomatized as follows:

$$\begin{array}{ll} & \forall x\ \neg\mathsf{atom}(x) \Rightarrow \mathsf{cons}(\mathsf{car}(x), \mathsf{cdr}(x)) = x \\ \forall x, y\ \mathsf{car}(\mathsf{cons}(x, y)) = x & \forall x, y\ \neg\mathsf{atom}(\mathsf{cons}(x, y)) \\ \forall x, y\ \mathsf{cdr}(\mathsf{cons}(x, y)) = y & \mathsf{atom}(\mathsf{nil}) \end{array}$$

$\boxed{T_\ell}$ is the theory that gives the axioms for the function length $\ell : \text{LISTS} \to \text{AG}$ and the constant $(1 : \text{AG})$: $\ell(\mathsf{nil}) = 0$; $\forall x, y\ \ell(\mathsf{cons}(x, y)) = \ell(y) + 1$; $1 \neq 0$

Applying some standard reasoning (see, e.g. [18]), we can substitute T_L with the set of the purely equational axioms of T_L, say $T_{L'}$, and enrich a bit the set of literals G to a set of literals G' in such a way $T_L \cup T_\ell \cup AG \cup G$ is equisatisfiable to $T_{L'} \cup T_\ell \cup AG \cup G'$. Let us choose as ordering an RPO with a total precedence $\succ$ such that all the symbols have a lexicographic status, except $+$, whose status is multiset, and such that it respects the following requirements: (a) $\mathsf{cons} \succ \mathsf{cdr} \succ \mathsf{car} \succ c \succ e \succ \ell$ for every constant c of sort LISTS and every constant e of sort ELEM; (b) $\ell \succ g \succ - \succ + \succ 0$ for every constant g of sort AG.

Proposition 4 *For any set G of ground literals, any saturation of $T_{L'} \cup T_\ell \cup G'$ w.r.t. $\mathcal{SP}_{AG}$ is finite.*

Trees with Size Let us reason about trees and their size. We can propose a formalization in which we need to reason about a theory of the kind $T_T \cup T_{size} \cup AG$, where T_T rules the behaviour of the trees and T_{size} constraints the behaviour of a function that returns the number of nodes of a tree. Thus we have:

$\boxed{T_T}$ has the mono-sorted signature $\Sigma_T := \{\mathcal{E} : \text{TREES}, \mathsf{binL} : \text{TREES} \to \text{TREES}, \mathsf{binR} : \text{TREES} \to \text{TREES}, \mathsf{bin} : \text{TREES} \times \text{TREES} \to \text{TREES}\}$, and it is axiomatized as follows:

$$\forall x, y\ \mathsf{binL}(\mathsf{bin}(x,y)) = x \qquad \forall x, y\ \mathsf{binR}(\mathsf{bin}(x,y)) = y$$
$$\forall x\ \mathsf{bin}(\mathsf{binL}(x), \mathsf{binR}(x)) = x$$

$\boxed{T_{size}}$ is the theory that gives the axioms for the function $\mathsf{size} : \text{TREES} \to \text{AG}$: $\mathsf{size}(\mathcal{E}) = 0$; $\forall x, y\ \mathsf{size}(\mathsf{bin}(x,y)) = \mathsf{size}(x) + \mathsf{size}(y)$

Let us now put as ordering an RPO with a total precedence $\succ$ on the symbols of the signature such that all the symbols have a lexicographic status, except $+$, whose status is multiset, and such that it respects the following requirements: (a) $\mathsf{bin} \succ \mathsf{binR} \succ \mathsf{binL} \succ c \succ \mathsf{size}$ for every constant c of sort TREES; (b) $\mathsf{size} \succ g \succ - \succ + \succ 0$ for every constant g of sort AG.

Proposition 5 *For any set G of ground literals, any saturation of $T_T \cup T_{size} \cup G$ w.r.t. $\mathcal{SP}_{AG}$ is finite.*

Application *(Algorithm 2.8 in [25]: Left-Rotation of trees)* Using the procedure induced by the calculus $\mathcal{SP}_{AG}$, it is possible to verify, e.g. that the input tree x and the output tree y have the same size:

1. $t := x$; 2. $y := \mathsf{binR}(t)$; 3. $\mathsf{binR}(t) := \mathsf{binL}(y)$; 4. $\mathsf{binL}(y) := t$; 5. Return y

In order to check that the size of x is exactly the one of y, we check for unsatisfiability modulo $T_T \cup T_{size} \cup AG$ the following constraint (see, again [25]):

$$\mathsf{binR}(t') = \mathsf{binL}(\mathsf{binR}(x')) \wedge \mathsf{binL}(t') = \mathsf{binL}(x') \wedge \mathsf{binL}(y') = t'$$
$$\wedge\, \mathsf{binR}(y') = \mathsf{binR}(\mathsf{binR}(x')) \wedge \mathsf{size}(x') \neq \mathsf{size}(y')$$

where x', y' and t' are fresh constants that identify the trees on which the algorithm applies.

6.1 Applying the Combination Framework

In the section above we have shown some examples of theories that extend the theory of abelian groups and whose constraint satisfiability problem is decidable. We have proved that AG can be enlarged to AG^* and AG and AG^* behave the same w.r.t. the satisfiability of constraints; moreover we have checked that AG is a Noetherian theory. To guarantee now that the theories that have been studied can be combined together it is sufficient to show that they fully satisfy the requirement of being AG-compatible and effectively Noetherian extension of AG (requirements (2a) and (2b) of Theorem 1). The AG-compatibility both of lists with length and trees with size is easily ensured observing that a constraint is satisfied w.r.t. $T_L \cup T_\ell \cup AG$ iff it is satisfied w.r.t. $T_L \cup T_\ell \cup AG^*$ and, analogously, any constraint is satisfiable w.r.t. $T_T \cup T_{size} \cup AG$ iff it is w.r.t. $T_T \cup T_{size} \cup AG^*$.

Moreover, checking the shape of the saturations produced, it is immediate to see that all the hypotheses required by Proposition 3 are satisfied when considering both the cases of lists with length and trees with size, turning $\mathcal{SP}_{AG}$ not only into a decision procedure for the constraint satisfiability problem, but also into an effective method for deriving complete sets of logical consequences over the signature of abelian groups (namely, the AG-bases). This implies that also the requirement (2b) of being effectively Noetherian extensions of abelian groups is fulfilled for both lists with length and trees with size. To sum up, we have proved that the theories presented so far can be combined preserving the decidability of the constraint satisfiability problem.

7 Conclusion

The problem of integrating a reasoning modulo arithmetic properties into the superposition calculus has been variously studied, and different solutions have been proposed, both giving the possibility of reasoning modulo the linear rational arithmetic ([15]) and relying on an over-approximation of arithmetic via abelian groups ([12, 22]) or divisible abelian groups ([23, 24]).

We have focused on the second kind of approach, giving an original solution to the satisfiability problem in combinations of theories sharing the theory of abelian groups. We have shown that in this case all the requirements to apply the non-disjoint combination method are satisfied, and we have considered an appropriate superposition calculus modulo abelian groups in order to derive satisfiability procedures. This calculus relies on a non trivial adaptation the one proposed in [12]: We consider a many-sorted and constraint-free version of the calculus, in which we use a restricted form of unification in abelian groups with free symbols, and in which only literals are involved. Under these assumptions we have proved that the calculus is refutationally complete, but, as a side remark, we notice that the same kind of proof works also in case the rules are extended to deal with Horn clauses and also, exactly as it happens in [12], after the introduction of an appropriate rule for the Factoring, to deal with general clauses. Our focus on the unit clause case is justified by our interest in the application to particular theories whose formalization is actually through axioms of that form.

It is worth noticing that two combination methods are involved in our approach: the method for unification problems [3] and the non-disjoint extension of Nelson-Oppen for satisfiability problems [11].

The framework for the non-disjoint combination used here cannot be applied, as it is, to the case where we consider a combination of theories sharing the Presburger arithmetic, because the latter is not Noetherian. Another framework, able to guarantee the termination of the resulting procedure on all the inputs, should be designed for that case.

We envision several directions for future work. As a first direction, we would like to relax current restrictions on theories and saturation types to apply effectively the calculus in the non-disjoint combination method. At the moment, since the presence of variables of sort AG into the clauses is not allowed, the results in [18] are not subsumed by the present paper. That restriction is justified by technical reasons: an important issue would be to discard it, enlarging in this way the applicability of our results. As a second direction we foresee, it would be interesting to find general methods to ensure the termination of the calculus by developing, for instance, an automatic meta-saturation method [16], or by considering a variable-inactivity condition [1]. Finally, it would be interesting to study how our calculus can be integrated into Satisfiability Modulo Theories solvers, by exploiting for instance the general framework developed in [4].

References

[1] A. Armando, M. P. Bonacina, S. Ranise, and S. Schulz. New results on rewrite-based satisfiability procedures. *ACM Transactions on Computational Logic*, 10(1), 2009.

[2] A. Armando, S. Ranise, and M. Rusinowitch. A rewriting approach to satisfiability procedures. *Information and Computation*, 183(2):140–164, 2003.

[3] F. Baader and K. U. Schulz. Unification in the union of disjoint equational theories: Combining decision procedures. *Journal of Symbolic Computation*, 21(2):211–243, 1996.

[4] M. P. Bonacina and M. Echenim. T-decision by decomposition. In *Proc. of CADE'07*, volume 4603 of *LNCS*, pages 199–214. Springer, July 2007.

[5] M. P. Bonacina, S. Ghilardi, E. Nicolini, S. Ranise, and D. Zucchelli. Decidability and undecidability results for Nelson-Oppen and rewrite-based decision procedures. In *Proc of IJCAR'06*, volume 4130 of *LNCS*, pages 513–527. Springer, 2006.

[6] A. Boudet, J.-P. Jouannaud, and M. Schmidt-Schauß. Unification in boolean rings and abelian groups. In C. Kirchner, editor, *Unification*, pages 267–296. Academic Press, London, 1990.

[7] P. L. Chenadec. *Canonical Forms in Finitely Presented Algebras*. Research Notes in Theoretical Computer Science. Pitman-Wiley, 1986.

[8] L. M. de Moura and N. Bjørner. Engineering dpll(t) + saturation. In *Proc. of IJCAR'08*, volume 5195 of *LNCS*, pages 475–490. Springer, 2008.

[9] N. Dershowitz. Orderings for term-rewriting systems. *Theoretical Computer Science*, 17(3):279–301, 1982.

[10] P. C. Eklof and G. Sabbagh. Model-completions and modules. *Annals of Mathematical Logic*, 2:251–295, 1971.

[11] S. Ghilardi, E. Nicolini, and D. Zucchelli. A comprehensive combination framework. *ACM Transactions on Computational Logic*, 9(2):1–54, 2008.

[12] G. Godoy and R. Nieuwenhuis. Superposition with completely built-in abelian groups. *Journal of Symbolic Computation*, 37(1):1–33, 2004.

[13] M. J. Hall. *The Theory of Groups.* New York The Macmillan Company, 1968.

[14] W. Hodges. *Model Theory.* Number 42 in Encyclopedia of Mathematics and its Applications. Cambridge University Press, 1993.

[15] K. Korovin and A. Voronkov. Integrating linear arithmetic into superposition calculus. In *Proc. of CSL'07*, volume 4646 of *LNCS*, pages 223–237. Springer, 2007.

[16] C. Lynch and D.-K. Tran. Automatic Decidability and Combinability Revisited. In *Proc. of CADE'07*, volume 4603 of *LNCS*, pages 328–344. Springer, 2007.

[17] G. Nelson and D. C. Oppen. Simplification by cooperating decision procedures. *ACM Transaction on Programming Languages and Systems*, 1(2):245–257, 1979.

[18] E. Nicolini, C. Ringeissen, and M. Rusinowitch. Satisfiability procedures for combination of theories sharing integer offsets. In *Proc. of TACAS'09*, volume 5505 of *LNCS*, pages 428–442. Springer, 2009.

[19] R. Nieuwenhuis and A. Rubio. Paramodulation-based theorem proving. In A. Robinson and A. Voronkov, editors, *Handbook of Automated Reasoning*, volume I, chapter 7, pages 371–443. Elsevier Science, 2001.

[20] G. E. Peterson and M. E. Stickel. Complete sets of reductions for some equational theories. *J. ACM*, 28(2):233–264, 1981.

[21] G. Plotkin. Building-in equational theories. *Machine Intelligence*, 7:73–90, 1972.

[22] J. Stuber. Superposition theorem proving for abelian groups represented as integer modules. *Theoretical Compututer Science*, 208(1-2):149–177, 1998.

[23] U. Waldmann. Superposition and chaining for totally ordered divisible abelian groups. In *Proc. of IJCAR'01*, volume 2083 of *LNCS*, pages 226–241. Springer, 2001.

[24] U. Waldmann. Cancellative abelian monoids and related structures in refutational theorem proving (Part I,II). *Journal of Symbolic Computation*, 33(6):777–829, 2002.

[25] T. Zhang. *Arithmetic integration of decision procedures.* PhD thesis, Department of Computer Science, Stanford University, Stanford (U.S.), 2006.

A Unification in Abelian Groups

In this section, we rely on the combination algorithm for unification problems developed by Baader-Schulz, which makes use of unification of linear constant restriction [3]. Strictly speaking, we use a (straightforward) many-sorted extension of the Baader-Schulz combination algorithm, to deal with unification problems where function symbols in AG and free function symbols share the sort AG. Given a unification problem Γ, a set of free constants C, and $<$ a linear constant restriction over C, (Γ, C) denotes the E-unification problem Γ with free constants C and $(\Gamma, C, <)$ denotes the E-unification problem (Γ, C) with the linear constant restriction $<$ over C. A complete set of E-unifiers of (Γ, C) (resp. $(\Gamma, C, <)$) is denoted by $CSU_E(\Gamma, C)$ (resp. $CSU_E^<(\Gamma, C)$).

The results we are using for AG-unification with free symbols over the shared sort AG are consequences of more general results stated in the following for E-unification with free symbols over the shared sort s_0.

Definition 6 *Let Σ be a mono-sorted signature over the sort s_0 and let E be an equational Σ-theory. Let Σ' be a many-sorted signature such that s_0 is the unique symbol shared by Σ and Σ', and let F be the empty equational Σ'-theory. A general E-unification problem is an $E \cup F$-unification problem. A general E-ground unification problem is a general E-unification problem in which no variable of sort s_0 occurs.*

Lemma 3 *Let $(\Gamma, C, <)$ be an E-unification problem with linear constant restriction $<$, such that Γ is of the form $\bigwedge_{i=1}^n x_i = t_i$, where t_i is ground for $i = 1, \ldots, n$. If E admits a convergent TRS, then it is possible to construct a $CSU_E^<(\Gamma, C)$ which is either empty or a singleton whose unique element is a ground substitution.*

Proof. The repeated application of the rules given below allows us to obtain a solved form of the E-unification problem with constants (Γ, C).

1. $\Gamma \wedge x = s \wedge x = t \vdash \Gamma \wedge x = s \downarrow_R$ if $x \in Var(\Gamma) \backslash C$ and $s \downarrow_R = t \downarrow_R$
2. $\Gamma \wedge x = s \wedge x = t \vdash \perp$ if $x \in Var(\Gamma) \backslash C$ and $s \downarrow_R \neq t \downarrow_R$
3. $\Gamma \wedge x = s \vdash \Gamma$ if $x \in C$ and $x = s \downarrow_R$
4. $\Gamma \wedge x = s \vdash \perp$ if $x \in C$ and $x \neq s \downarrow_R$

If the solved form is $\perp$, then $CSU_E^<(\Gamma, C) = \emptyset$. Otherwise, the solved form corresponds to a grounding substitution in R-normal form, say σ. To obtain solutions satisfying the linear constant restriction $<$, we still have to eliminate a constant c from the term $x\sigma$ if $x < c$, which means that we have to find a substitution μ such that there exists a term u satisfying $x\sigma\mu =_E u$ and $c \notin u$. Since $x\sigma$ is ground, μ is the identity, and a term u with $c \notin u$ exists iff $c \notin x\sigma$. Indeed, rewriting w.r.t R does not introduce new free constants, and so $c \notin u$ implies $c \notin u \downarrow_R = x\sigma \downarrow_R = x\sigma$. Consequently, $CSU_E^<(\Gamma, C) = \{\sigma\}$ if

$$\forall x \in Var(\Gamma) \backslash C \ \forall c \in C, \ x < c \Rightarrow c \notin x\sigma.$$

Otherwise, $CSU_E^<(\Gamma, C) = \emptyset$.

Lemma 4 *If E is an equational theory admitting a convergent TRS, then general E-ground unification is finitary, and for any general E-ground unification problem Γ, we have: $\forall \sigma \in CSU_{E \cup F}(\Gamma), VRan(\sigma) \subseteq Var(\Gamma)$.*

Proof. The Baader-Schulz combination algorithm for unification [3] can be applied to solve an $E \cup F$-unification problem Γ in which no variable of sort s_0 occurs. First, we purify the equations of Γ by applying repeatedly the following rules:

VA: $\Gamma \wedge s[u]_\omega = t \vdash \Gamma \wedge s[x]_\omega = t \wedge x = u$ if u is a direct alien subterm of s occurring at position ω, and x is new variable (of sort s_0)

PurifEq: $\Gamma \wedge s = t \vdash \Gamma \wedge x = s \wedge x = t$ if s, t are pure (non-variable) terms, and x is a new variable (of sort s_0)

This purification process applied on Γ terminates and leads to an $E \cup F$-unification problem $\Gamma_E \wedge \Gamma_F$, such that Γ_E and Γ_F are respectively E-pure and F-pure. Then, the combination algorithm considers all possible cases of

- partitions of the set of shared variables V into $V_E \oplus V_F$,
- identifications ξ of $V_E \oplus V_F$ (i.e. idempotent substitutions whose domains and ranges are both included in $V_E \oplus V_F$) such that $x\xi = y\xi$ implies $x, y \in V_E$ or $x, y \in V_F$,
- linear orderings $<$ over $V_E\xi \oplus V_F\xi$

and calls unification algorithms with linear constant restriction known for E and F to solve the respective inputs $(\Gamma_E\xi, V_F\xi, <)$ and $(\Gamma_F\xi, V_E\xi, <)$.

Since F is the empty theory, it is sufficient to consider the linear constant restrictions such that:

(i) If $x = t$ occurs in Γ_F, then $x \in V_F$

(ii) If $x = t$ occurs in Γ_F and $c \in t$ for some $c \in V_E$, then $c < x$

since for the other linear constant restrictions, one get F-unification problems with no solutions. Thanks to (i), all abstraction variables occurring in Γ_E (introduced during the purification process) are necessarily in V_F, and so are considered as free constants in $(\Gamma_E\xi, V_F\xi, <)$. Therefore, Lemma 3 can be applied, and for each solution $\sigma_E \in CSU_E^{<}(\Gamma_E\xi, V_F\xi)$ there are no new variables introduced. It is well-known that the same property holds also for the empty theory F and any solution $\sigma_F \in CSU_F^{<}(\Gamma_F\xi, V_E\xi)$. The set of all conjunctions of (the solved forms of) ξ, σ_E and σ_F represents a disjunction of "dag" solved forms and after variable replacement we obtain a disjunction of solved forms equivalent to $\Gamma_E \wedge \Gamma_F$. The restrictions of these solved forms to $Var(\Gamma)$ define a complete set of $E \cup F$-unifiers of Γ satisfying the required property.

Lemma 5 *Let Γ be a E-unification problem, and let $CSU_E(\Gamma)$ be a complete set of E-unifiers of Γ such that $\forall \mu \in CSU_E(\Gamma), VRan(\mu) \subseteq Var(\Gamma)$. For any E-unifier σ of Γ such that $Dom(\sigma) = Var(\Gamma)$, there exists $\mu \in CSU_E(\Gamma)$ such that $\sigma =_E \mu(\sigma_{|VRan(\mu)})$.*

Proof. By definition of a complete set of E-unifiers, there exist $\mu \in CSU_E(\Gamma)$ and a substitution σ' such that $\forall x \in Var(\Gamma)$, $x\sigma =_E x\mu\sigma'$. By definition, $VRan(\mu) \cap Dom(\mu) = \emptyset$, and so $\forall x \in VRan(\mu)$, $x\mu = x$. Since $VRan(\mu) \subseteq Var(\Gamma)$, we have: $\forall x \in VRan(\mu)$, $x\sigma =_E x\mu\sigma' = x\sigma'$ which means that $\sigma_{|VRan(\mu)} =_E \sigma'_{|VRan(\mu)}$. Hence, $\sigma_{|VRan(\mu)}$ can replace $\sigma'_{|VRan(\mu)}$ to instantiate terms in the range of μ and so we get (1) $\forall x \in Var(\Gamma)$, $x\sigma =_E x\mu(\sigma_{|VRan(\mu)})$. By assumption, $Var(\Gamma) = Dom(\sigma)$, and both $Dom(\mu)$ and $VRan(\mu)$ are included in $Var(\Gamma)$. Consequently, we have that (2) for any variable x not in $Var(\Gamma)$, $x\sigma = x = x\mu(\sigma_{|VRan(\mu)})$. Finally, (1) and (2) imply that $\sigma =_E \mu(\sigma_{|VRan(\mu)})$.

Lemma 1 directly follows from Lemma 4 and Lemma 5.

B Amalgamation Property of *AG*

In this section we prove that the theory of abelian groups *AG* has the amalgamation property. Though the result should be known, we are not able to provide a precise reference where the property is completely illustrated. For that reason, we prefer here to propose an argument by ours, moving from a basic construction in [13].

Theorem 2. *AG has the amalgamation property.*

Proof. The amalgamation property for abelian groups can be restated as follows: let H, K and A' be s.t. there exist two embeddings $i : A' \to H$ and $\iota : A' \to K$. We want to show that there exists a group L such that:

1. L is abelian;
2. there exist two embeddings of abelian groups $f : H \to L$ and $g : K \to L$ such that the composition are equal, i.e. $i \circ f = \iota \circ g$.

Since i and ι are embeddings from A' to H and K respectively, it means that there exist a subgroup of H, let us say A, and a subgroup of K, let us say B, such that $A = i(A')$, $B = \iota(A')$, and such that A and B are isomorphic (the isomorphism is given by $i^{-1} \circ \iota$, or $\iota^{-1} \circ i$). In order to find the subgroup L needed, we can start moving from H and K, adapting a case that can be found in [13].

Building L Let S be the set of all the finite words over the alphabet $H \cup K$ and let $\sim$ be the smallest equivalence relation on words such that, for all the words α, β:

E1 if 1 is 1_H or 1_K, $\alpha 1 \beta$ is equivalent to $\alpha\beta$;

E2 if h_1 and h_2 belong both to H and in H $h_1 h_2 = \hat{h}$,[2] then $\alpha h_1 h_2 \beta$ is equivalent to $\alpha\hat{h}\beta$;

E3 if k_1 and k_2 belong both to K and in K $k_1 k_2 = \hat{k}$, then $\alpha k_1 k_2 \beta$ is equivalent to $\alpha\hat{k}\beta$;

E4 if a is an element of A and b is an element of B such that $b = \iota(i^{-1}(a))$, or, equivalently, $a = i(\iota^{-1}(b))$, then $\alpha a \beta$ is equivalent to $\alpha b \beta$;

[2]For sake of readability, we will not be very precise in using different symbols for the operation of concatenation of words over a given alphabet and the product of elements in a group: the context should be enough to avoid ambiguities.

E5 if h is an element of H and k is an element of K, then $\alpha hk\beta$ is equivalent to $\alpha kh\beta$.

We can consider the quotient of S with respect to $\sim$: $S/\sim$, and we can introduce a binary operation $*$ over the equivalence classes in $S/\sim$, relying on the concatenation: if $[f_1]$ and $[f_2]$ are equivalence classes, then $[f_1]*[f_2] := [f_1 f_2]$. It is easy to verify that

- the product $*$ is well-defined: if $f_1 \sim f_1'$ and $f_2 \sim f_2'$, then $[f_1 f_2] = [f_1' f_2']$;
- the product $*$ is associative;
- the (class of the) void word is the identity of the product;
- if $f = g_1 \dots g_n$, then $f^{-1} := g_n^{-1} \dots g_1^{-1}$ is the representative of the inverse of $[f]$ (being the g_i's simply letters of the alphabet $H \cup K$).

So, let us name L the group $L := \langle S/\sim, *\rangle$.

L is commutative: by induction, it is easy to see that, if $f_1 = g_1 \dots g_n$, $f_2 = g_1' \dots g_m'$, $[f_1] * [f_2] = [f_2] * [f_1]$.

Building f and g Our aim is now to build the appropriate embeddings of H and K into L. To this aim, we will rely on the fact that in L can be defined a *normal form* of the equivalence classes.

We will call the algorithm for the reduction of a word in its normal form a word process. Before starting to describe it, we introduce the definition of a word in canonical form. Relying on the decomposition of a group in right cosets of a subgroup[3], we decompose H into right cosets of A, choosing the identity of H: 1_H as the representative of A, and we decompose K into right cosets of B, choosing the identity of K: 1_K as the representative of B. Thus $H = A + Ah_1 + \cdots + Ah_p + \dots$, varying p into a set of indexes P, and $K = B + Bk_1 + \cdots + Bk_j + \dots$, varying j into a set of indexes J. Since B is the isomorphic image of A' through ι and A is the isomorphic image of A' through i, we shall write $H = A' + A'h_1 + \cdots + A'h_p + \dots$ and $K = A' + A'k_1 + \cdots + A'k_j + \dots$. Thus, every element in H as a unique representation as $h = ah_p$, where $a \in A'$ and h_p is one of the h's that are representatives of the cosets, and, analogously, every element in K has a unique representation as $k = ak_j$, where $a \in A'$ and k_j is one of the representatives (of course, if h or k is identified with an element in A', then it will be represented -uniquely- with the corresponding $a \in A'$).

We say that an element of L is in *canonical form* if it is of the form $l = ah_p k_j$, where $a \in A'$, $h_p \in H$, $k_j \in K$ and h_p and k_j are the representatives of the cosets.

We are now ready to define a word process (that is, a set of operation that allows us to reduce a given word in L to its canonical form).

Let $l = g_1 g_2 \dots g_t$, being the g_i's elements of H or K; we define a word process as follows:

$W_0 := \varepsilon$ (the empty word)

$W_1 :=$

- 1 if g_1 is the identity of H or K;

[3] If g is an element of the group G and N is a subgroup of G, the right coset Ng is $\{ng \mid n \text{ is an element of } N\}$.

- ah if $g_1 \in H$, $g_1 = ah$ and ah is the presentation of g_1 in the decomposition of H through the right cosets of A';
- ak if $g_1 \in K$, $g_1 = ak$ and ak is the presentation of g_1 in the decomposition of K through the right cosets of A';
- a if $g_1 \in A'$;

Notice that W_1 is in canonical form. Suppose now that W_i is in canonical form: $W_i = ah_ik_j$.

W_{i+1} :=

- W_i if g_{i+1} is the identity of H or K;
- $\overline{a}h_ik_j$ where $\overline{a} = aa'$, if $g_{i+1} = a'$, $a' \in A'$;
- $\hat{a}h^*k_j$ if $g_{i+1} = a'h'$, $a'h'$ is the presentation of g_{i+1} in the decomposition of H through the right cosets of A', a^*h^* is the presentation of h_ih' in the decomposition of H through the right cosets of A' and $\hat{a} = aa'a^*$;
- $\tilde{a}h_ik^{**}$ if $g_{i+1} = a''k''$, $a''k''$ is the presentation of g_{i+1} in the decomposition of K through the right cosets of A', $a^{**}k^{**}$ is the presentation of k_jk'' in the decomposition of K through the right cosets of A' and $\tilde{a} = aa''a^{**}$.

By construction, it is clear that, for every element $l = g_1g_2 \dots g_t$ in L, the word process halts at step t, returning a word in canonical form. Moreover, since at every step i it holds that $W_ig_{i+1} \sim W_{i+1}$, l and W_t belong to the same equivalence class. Finally, by induction it is easy to prove that, if l_1 and l_2 are words belonging to the same equivalence class, they have the same canonical form, and if l is in canonical form, then the word process leaves l unchanged.

Collecting everything together, we obtain that, once the cosets representatives have been chosen, in each equivalence class of elements in L there is one, and only one, element in canonical form: $l = ahk$, where $a \in A'$, h, k are coset representatives in H and K, respectively, different from the unity $1_{A'}$ of A' and taken from some arbitrary but fixed selection of coset representatives.

Now, let us define $f : H \to L$ as the map that associates, to each element $h \in H$, the class $a'h_i$ in L, where $h = a'h_i$ is the decomposition by coset representatives of h in $H = A' + A'h_1 + \cdots + A'h_i + \dots$.

f is clearly an homomorphism between groups; it is also injective because different elements have a different decomposition through coset representatives. These two properties exactly mean that f is an embedding of H into L.

Analogously, we can define $g : K \to L$ as the map that associates, to each element $k \in K$, the class $a'k_j$ in L, where $h = a'k_j$ is the decomposition by coset representatives of k in $K = A' + A'k_1 + \cdots + A'k_j + \dots$, and again f is an embedding of K into L.

Let us now consider an element $a \in A'$. a will be mapped into $i(a)$ by the map i from A to H, and will be mapped by f into itself. On the other hand, a will be mapped into the element $\iota(a)$, that will be mapped by g again into a. In other words, both $i \circ f$ and $\iota \circ g$ act as the identity map on A'.

Thus we have proved there exist two embeddings $f : H \to L$ and $g : K \to L$ such that $i \circ f = \iota \circ g$.

C Refutational Completeness of $\mathcal{SP}_{AG}$

Let us start proving the following proposition:

Proposition 2. *Let s be a term in which no variable of sort* AG *occurs, let σ be a grounding substitution such that both s and σ are in AG-normal form, and let R be a ground TRS such that (s, σ) is irreducible w.r.t. R. Moreover, let $\sigma =_{AG} \mu\pi$, where π is another grounding substitution in AG-normal form and μ is a substitution that does not have variables of sort* AG *in its range. Then $(s\mu, \pi)$ is still irreducible w.r.t. R.* *Proof.* Of course, AG-nf$(s\sigma)$ is equal to AG-nf$(s\mu\pi)$, and maxred$_R(s\sigma)$ is equal to maxred$_R(s\mu\pi)$.

If s is a variable, since (s, σ) is irreducible, it follows that $s\sigma$ is R-irreducible, and so also $(s\mu, \pi)$ is irreducible.

Suppose now that s is not a variable. Since the presence of variables of sort AG is forbidden, variables can occur in (proper) subterms of s of the kind $f(x, t_1, \ldots, t_n)$, for f different from $+$ and $-$. Let us focus our attention over such variables. We have only two cases to consider:

- $x\sigma$ is R-irreducible, and so the pair $(x\mu, \pi)$ is irreducible. This implies that, when unfolding the definition of irreducibility, all the subterm of the kind $x\mu$ in $s\mu$ will satisfy, whenever needed, the requirement for the irreducibility of $(s\mu, \pi)$;
- $x\sigma$ is R-reducible. Since (s, σ) is irreducible, it means that the occurrences of $x\sigma$ are only in subterms that are deleted during the reduction in AG-normal form of $s\sigma$, thus implying that all the terms of the kind $x\mu$ have no influence in the check of the irreducibility of $(s\mu, \pi)$.

Sometimes, it will be useful also the notion of irreducibility related to a certain term u:

Definition 7 *Let s be a term in which no variable of sort* AG *occurs, let σ be a grounding substitution, let both s and σ be in AG-normal form, let u be a term and let R be a ground TRS. The pair (s, σ) is $(u, \succeq)$-irreducible:*

- *if s is a variable, either $u \prec s\sigma$, or $u \succeq s\sigma$ and $s\sigma$ is R-irreducible;*
- *if s is not a variable, for for each term $t = f(t_1, \ldots, t_n)$ such that s is in the form $t + v$ or $-t + v$ and such that $u \succeq$ AG-nf$(t\sigma)$, each (t_i, σ) is irreducible.*

Before stating completely the proof of the refutational completeness of $\mathcal{SP}_{AG}$, we recall the following Lemma (its proof can be found in [12], but we restate it here for sake of completeness).

Lemma 6 *Let M_{red} be the reductive form of some literal $M\sigma$ in $Ir_{R_S}(S)$ such that $I \not\models M\sigma$; let M_{red} be not in the form $t \neq t$, and let s be the maximal summand in M_{red}. M_{red} is either (a) in the form $ms = t$, with $s \succ t$, or (b) $ms \neq t$, with $s \succeq t$. In both cases, ms is reducible by R_S.*

Proof. Indeed, suppose (a) holds. Since $I \not\models ms = t$, M_{red} has generated no rule of R_S, thus, according to Definition 5, the only possibility is that ms is already reducible by $R_{M_{red}}$. Suppose, on the other hand, that (b) holds. The fact that $I \not\models ms \neq t$ means exactly that $I \models ms = t$, and so ms and t are joinable by $R_S \cup R_{AG}$, $ms \succ t$ and, since ms is in AG-normal form and is the maximal side, ms is reducible by R_S.

Lemma 2. *Let S be the closure under $\mathcal{SP}_{AG}$ of a set of literals S_0, and let us assume that the empty clause does not belong to S. Let I be the model of terms derived from S as described in Section 5, and let $Ir_{R_S}(S)$ be the set of ground instances $L\sigma$ in S such that (L, σ) is irreducible w.r.t. R_S. Then:*

1. *$I \models Ir_{R_S}(S)$ implies that $I \models S$.*
2. *$I \models Ir_{R_S}(S)$.*

Proof.

1. ([12]). For each ground instance $L\sigma$ of a literal L in S, let us consider an other instance $L\sigma'$ of L, where $x\sigma'$ is the normal form of $x\sigma$ w.r.t. R_S for every variable x of L. Naturally, $L\sigma'$ is an other instance of S that is in $Ir_{R_S}(S)$. Since by hypothesis $I \models Ir_{R_S}(S)$, it follows that $I \models L\sigma'$, which implies, due to the definition of the congruence on I, that $I \models L\sigma$. In other words, we have proved that, for each ground instance $L\sigma$ of a literal L in S, $I \models L\sigma$, which immediately leads to the conclusion that $I \models S$.

2. The strategy here is to derive a contradiction from the existence of an irreducible literal that is not verified in I. In order to produce this contradiction, we rely on the technique presented in [12] and we adapt it to our context.

   Let M_{red} be the minimal, w.r.t. $\succ$, literal that is the reductive form of some $M\sigma$ in $Ir_{R_S}(S)$ such that $I \not\models M_{red}$, and suppose that M is in the form $e \bowtie t'$. If the literal M_{red} is in the form $t \neq t$, then an application of the rule Reflection to M could have been possible, thus producing the empty clause in S.

   Otherwise, we will show that it is possible to perform some inferences involving M and producing a new literal to which it will be possible to apply the following schema of reasoning:

   $\boxed{\text{Proof Pattern}}$ Assume that (i) there exists a literal P such that (P, σ) is irreducible and such that it admits an orientation of the kind $l = r$, and (ii) there exists a position q in e such that $e|_q$ is a non variable term that is not immediately below a $+$ or a $-$ and that admits a splitting $e'_1 + e'_2$, and that (iii) the following inferences are admissible:

   $$\frac{l = r \qquad e[e'_1 + e'_2]_q \bowtie t'}{(e[r + e'_2]_q \bowtie t')\mu_i}$$

   where μ_i ranges in the complete set of unifiers for the unification problem $l =_{AG} e'_1$, and where $l\sigma =_{AG} e'_1\sigma$. Moreover, assume that (iv) $(AG\text{-nf}(e[r + e'_2]_q) \bowtie t', \sigma)$ is irreducible and that (v) the AG-normal form of $(e[r+e'_2]_q \bowtie t')\sigma$ is smaller than M_{red} and not satisfied in I.

   At this point, if $\sigma =_{AG} \mu_j \sigma_{|VRan(\mu_j)}$ is the decomposition of the substitution σ according to Lemma 1, from Proposition 2 it follows that also $((AG\text{-nf}(e[r+e_2']_q) \bowtie t')\mu_j, \sigma_{|VRan(\mu_j)})$ is irreducible, and relying on the confluence of the process of reduction in AG-normal form, it follows again that the reductive form of $(e[r+e_2']_q \bowtie t')\mu_j\sigma_{|VRan(\mu_j)}$ is smaller than M_{red} and it is still not verified in I.

Thus, every time we will be able to apply the proof pattern above, we will be able to prove the existence of an irreducible literal that is not verified in I and that is smaller than the minimal literal satisfying the same property, the wanted contradiction.

Let us enter now a little bit more into the details. From now on, we will strongly rely on some – very technical – lemmas that are stated in [12]. The extension to the many-sorted case is straightforward, so we will omit their proofs here, but we will quote them when needed. First of all, we focus more on the shape of M_{red}. Since it is not in the shape $t \neq t$, it can be in the form $ms = t$, with $s \succ t$, or $ms \neq t$, with $s \succeq t$. In both the cases, by Lemma 6, ms has to be reducible, and the rule reducing ms has to come from the reductive form P_{red} of some ground instance $P\sigma$ of an equation P in S. Since P_{red} has produced a rule, $P\sigma$ belongs to $Ir_{R_S}(S)$.

Moreover, adapting Lemma 44 of [12], it is possible to prove that there exists a subterm s' of ms such that: (i) if s' is in the form $nu + v$, then the rule $nu \rightarrow r'$ is in R_S for some term r', (ii) if s' is in the form $-u+v$, then the rule $-u \rightarrow (n-1)u + r'$ is in R_S for some term r', (iii) either s' is ms or s' occurs in ms in a position $p \cdot i$ such that the topmost symbol of $(ms)|_p$ is neither $+$ nor $-$. From the considerations above, it follows that no rule with a left-hand side containing a term bigger that u can reduce s'. It is possible now to distinguish two cases:

– *A) The rule reducing ms is $nu \rightarrow r'$.*

  In this case it is possible to prove (see Lemma 45 of [12]) that there exists an orientation $l = r$ of the equation P such that AG-nf$(l\sigma)$ is nu and AG-nf$(r\sigma)$ is r'. Moreover (r, σ) is $(u, \succeq)$-irreducible. At this point it is convenient to focus on two different possibilities:

  ∗ *A.1) s' is ms*

    Then s is u, so M_{red} could be rewritten into $mu \bowtie t$ for some $m \geq n$. We treat differently the case in which ms, t, nu and r' are of sort different from AG or of sort AG.

    · *A.1.1) ms, t, nu and r' are of sort different from* AG.

      This implies that $n = m = 1$ and that the literal $s \bowtie t$ is the instantiation of the literal $M \equiv e_1 \bowtie e_2$ through σ (and, if needed, some steps of AG-normalization). Since s is reducible by R_S and since (M, σ) is irreducible w.r.t. R_S, it implies that e_1 is not a variable, so the following inferences are possible:

$$\frac{l = r \qquad e_1 \bowtie e_2}{(r \bowtie e_2)\mu_i}$$

      where μ_i ranges in the complete set of unifiers for the unification problem $l =_{AG} e_1$. It is easy to check that $(r \bowtie e_2, \sigma)$ is irreducible, that $(r \bowtie e_2)\sigma$ is not verified in I and that its reduced form is smaller that M_{red}: at this point we can apply the proof pattern above, deriving the wanted contradiction.

· *A.1.2) ms, t, nu and r' are of sort* AG.
In this case M is in the form $e \bowtie 0$ and AG-nf$(e\sigma)$ is $mu - t$, for $m \geq n$. Moreover, u is the maximal summand of $ms - t$ and (e, σ) is, by hypothesis, irreducible w.r.t. R_S. It is possible now to re-adapt Lemma 47 of [12] and to find a splitting $e_1 + e_2$ of e such that $(e_1 + e_2)\sigma =_{AG} e\sigma$, $e_1\sigma$ is nu, (e_2, σ) is $(u, \succeq)$-irreducible and the maximal summand of $e_2\sigma$ is smaller or equal to u. So, the following inferences are allowed:
$$\frac{l = r \qquad e_1 + e_2 \bowtie 0}{(r + e_2 \bowtie 0)\mu_i}$$
where, as before, μ_i ranges in the complete set of unifiers for the unification problem $l =_{AG} e_1$. Again, it is easy to check that also $(r + e_2 \bowtie 0, \sigma)$ is irreducible, that $(r + e_2 \bowtie 0)\sigma$ is not verified in I and that its reduced form is smaller that M_{red}. Thus, as in the previous case, the proof pattern applies, deriving a contradiction.

∗ *A.2) s' is a proper subterm of ms*
More precisely, $ms|_p$ is s' for some position p below some s. From now on, we will denote with e the left-hand side of the literal M, that is in one-sided form, notwithstanding its sort. Then, reproducing the argument of Lemmas 50 and 51 in [12] it is possible to find a position q in e such that $e|_q =_{AG} s'$, $(e|_q, \sigma)$ is irreducible w.r.t. R_S and, for all the terms r'' in which no variable of sort AG occurs, $e[r'']_q\sigma =_{AG} ms[r''\sigma]_p$ in case ms is of sort different from AG, or $e[r'']_q\sigma =_{AG} ms[r''\sigma]_p - t$ in case ms is of sort AG. Moreover, if (r'', σ) is irreducible w.r.t. R_S and $s' \succ AG$-nf$(r''\sigma)$, then $(e[r'']_q, \sigma)$ is irreducible w.r.t. R_S. We notice that also $(e|_q, \sigma)$ is irreducible w.r.t. R_S and, recalling that s' is of the form $nu+s''$ and that u is the maximal among the reducible summands of s', it is possible to apply, properly adapted, Lemma 47 of [12], thus deriving the existence of a splitting $e'_1 + e'_2$ of $e|_q$ such that $(e'_1 + e'_2)\sigma =_{AG} e|_q\sigma$, $e'_1\sigma$ is nu, and (e'_2, σ) is $(u, \succeq)$-irreducible. Thus the following inferences are allowed:
$$\frac{l = r \qquad e[e'_1 + e'_2]_q \bowtie t'}{(e[r + e'_2]_q \bowtie t')\mu_i}$$
where, as before, μ_i ranges in the complete set of unifiers for the unification problem $l =_{AG} e_1$. The appropriate adaptation of Lemma 53 in [12] guarantees the irreducibility of $(e[r + e'_2]_q \bowtie t', \sigma)$; moreover, since $(e[r + e'_2]_q \bowtie t')\sigma$ is still not true in I and since its reduced form is smaller then M_{red}, we can apply again the proof pattern, obtaining the wished contradiction on the minimality of M_{red}.

– *B) The rule reducing ms is $-u \to (n-1) + r'$.*
Since M_{red} is the reductive form of $M\sigma$, s' cannot coincide with ms (it must be a proper subterm of s). At this point, the contradiction on the minimality of M_{red} follows from arguments very similar to the

ones applied in the previous case A.2), with a proper adaptation of Lemmas 46, 48, 51 and 54 in [12].

C.1 Computing AG-bases

Let us fix the following data: let $\Gamma(\underline{a}, \underline{b})$ be a set of ground literals over an expansion of $\Sigma \supseteq \Sigma_{AG}$ with the finite sets of fresh constants $\underline{a}, \underline{b}$, let $T \supseteq AG$ be a Σ-theory whose axioms are unit clauses. Suppose to perform a saturation w.r.t. $\mathcal{SP}_{AG}$ adopting an RPO ordering in which the precedence is $f \succ a \succ - \succ + \succ 0$ for every function symbol f in $\Sigma^{\underline{b}}$ different from $+, -, 0$, every constant a in $\underline{a}$ and that every symbol has a lexicographic status, except $+$, whose status is multiset. Proposition 3 shows how $\mathcal{SP}_{AG}$ can be used in order to derive AG-bases:

Proposition 3. *Let S be a finite saturation of $T \cup \Gamma$ w.r.t $\mathcal{SP}_{AG}$ not containing the empty clause and suppose that, in every equation $e = 0$ containing at least one of the constants a in $\underline{a}$ as summand, the maximal summand is not unifiable with any other summand in e. Then the set Δ of all the ground equations over $\Sigma^{\underline{a}}_{AG}$ in S is a AG-basis for T w.r.t. $\underline{a}$.*

Proof.

Since T is axiomatized by unit clauses, it is in particular a Horn theory, and so it is convex. Thus, when looking for AG-bases for T, it will be sufficient to focus simply on the ground equations over the sort AG that are implied by $T \cup \Gamma$. At this point, we have to prove that, if $e = 0$ is a ground equation over $\Sigma^{\underline{a}}_{AG}$ implied by $T \cup AG \cup \Gamma(\underline{a}, \underline{b})$, then $e = 0$ is already implied by $AG \cup \Delta$[4]. From the fact that $\mathcal{SP}_{AG}$ is refutational complete, we have that $T \cup AG \cup \Gamma(\underline{a}, \underline{b}) \models e = 0$ iff the saturation of $Ax(T) \cup \Gamma(\underline{a}, \underline{b}) \cup \{e \neq 0\}$ contains the empty clause, and thus iff the saturation of $S \cup \{e \neq 0\}$ contains the empty clause. Since S does not contain the empty clause, the only way to derive it is by reducing $e \neq 0$ by means of equations contained in S.

Let us start analyzing into details when we perform a reduction of $e \neq 0$ by an application of a direct superposition (the case of an application of an inverse superposition is very similar, and hence skipped). Thus, the maximal summand of e is of the kind ma for some integer $m > 0$ and some constant a in $\underline{a}$, and the only inferences needed are inferences with equations in S admitting an orientation of the kind $na = t$ for some positive integer n. The fact that the left-hand side of the orientation is already in the shape na is due to the fact that (i) the left-hand side of the orientation has to be unified with some subterm of ma, that is just a sum of constants, and that (ii) the presence of variables of sort AG is forbidden. Let us show, now that the term t can contain only constants. Suppose not; by the choice of the ordering, the maximal summand in $na = t$ is necessarily a term s different from a constant. s cannot be ground, otherwise no orientation with left-hand side na will ever be performed, being in this case always t bigger than na. So s could only be a non ground term, different from a variable. But now the hypothesis that s cannot unify with any of the other summands in the equation $na = t$ implies that, for every grounding θ, $s\theta$ is always the maximal summand, even after the usual step of

[4] As well as when performing the calculus, when dealing with literal we will always consider them in one-sided form.

AG-normalization. So, also in this case the inference with the orientation $na = t$ would be unnecessary in order to derive the empty clause.

Summing up, we have that $e \neq 0$ can be reduced using only equations in Δ, and any inference will produce new inequations $e' \neq 0$ that, applying the same arguments above, can be reduced by only equations in Δ.

Hence the saturation of $T \cup \Gamma(\underline{a}, \underline{b}) \cup \{e \neq 0\}$ does not contain the empty clause iff the saturation of $\Delta \cup \{e \neq 0\}$ does not contain it. Relying on the completeness of the calculus, we have obtained that $T \cup AG \cup \Gamma(\underline{a}, \underline{b}) \models e = 0$ iff $AG \cup \Delta \models e = 0$.

D Some examples

Lists with Length In order to prove its satisfiability of a set of ground literals G modulo $T_L \cup T_\ell \cup AG$ we first of all replace all the literals in $G \cup \{\mathsf{atom}(\mathsf{nil})\}$ in the form $\neg\mathsf{atom}(t)$ and $\mathsf{atom}(t')$ with respectively $t = \mathsf{cons}(sk_1, sk_2)$ and $\forall x_0, x_1\, t' \neq \mathsf{cons}(x_0, x_1)$, where t and t' are ground terms of sort LISTS and sk_1, sk_2 are fresh constants of the appropriate sort. Let now $T_{L'}$ be the subtheory of T_L whose axioms are just the first two (equational) axioms of T_L. We have (see e.g. [18]) that G is satisfiable w.r.t. $T_L \cup T_\ell \cup AG$ if and only if G' is satisfiable w.r.t. $T_{L'} \cup T_\ell \cup AG$. So, applying at most some standard steps of flattening, we can focus our attention to sets of literals of the following kind (x is a variable of sort ELEM, y is a variable of sort LISTS, $h, l, a, f, c, l_1, l_2, e, d, e_1, e_2, g$ are constants of the appropriate sorts), and the left-hand side of all the literals is the maximal one.

i.) equational axioms for lists
  a) $\mathsf{car}(\mathsf{cons}(x, y)) = x$;
  b) $\mathsf{cdr}(\mathsf{cons}(x, y)) = y$;

ii.) reduction for atom
  a) $\mathsf{cons}(x, y) \neq h$;
  b) $\mathsf{cons}(x, y) \neq \mathsf{nil}$;

iii.) axioms for the length
  a) $\ell(\mathsf{nil}) = 0$;
  b) $\ell(\mathsf{cons}(x, y)) = \ell(y) + 1$;

iv.) ground literals over the sort LISTS
  a) $\mathsf{cons}(e, l) = c$;
  b) $\mathsf{cdr}(f) = c$;
  c) $l_1 \bowtie l_2$;

v.) ground literals over the sort ELEM
  a) $\mathsf{car}(h) = d$;
  b) $e_1 \bowtie e_2$;

vi.) ground literals over the sort AG
  a) $ns \bowtie m_1t_1 + n_2t_2 + \cdots + m_nt_n$,

where the literals in the group vi are in reductive normal form, and s and t_i of the kind $\ell(a)$ or g (being the g's simply constants of sort AG).

Let us choose as ordering a RPO with a total precedence $\succ$ on the symbols of the signature such that all the symbols have a lexicographic status, except $+$, whose status is multiset, and such that respects the following requirements: (a) $\mathsf{cons} \succ \mathsf{cdr} \succ \mathsf{car} \succ c \succ e \succ \ell$ for every constant c of sort LISTS and every constant e of sort ELEM; (b) $\ell \succ g \succ - \succ + \succ 0$ for every constant g of sort AG.

These requirements over the precedence guarantee that every compound term of sort LISTS is bigger than any constant of the same sort, any compound term over the sort ELEM is bigger than any constant of sort ELEM, and that any term of the kind $\ell(a)$ is bigger than any constant of sort AG.

Proposition 4. *For any set G of ground literals, any saturation of $T_{L'} \cup T_\ell \cup G'$ w.r.t. $\mathcal{SP}_{AG}$ is finite.*

The key observations, in order to prove termination, are that the non-ground set of literals is already saturated, every (dis)equation obtained by the application of a rule to ground factors is smaller in the ordering w.r.t. the biggest factor in the antecedent of the rule, and every application of a rule of the calculus to a ground and a non-ground literal produces a ground literal that is smaller than the ground factor. In other terms, every literal produced during the saturation phase is ground and it is strictly smaller than the biggest ground literal in the input set. Since the ordering on the literals is the multiset extension of a terminating ordering, it is terminating too.

Trees with Size In order to check for satisfiability modulo $T_T \cup T_{size} \cup AG$ a set of ground literals G, we have to saturate a set of literals of the following kind (as usual, any set of ground literals can be seen, at most after the application of some standard steps of flattening, as a set of literals as below):

i.) equational axioms for trees

a) $\mathsf{binL}(\mathsf{bin}(x, y)) = x$;

b) $\mathsf{binR}(\mathsf{bin}(x, y)) = y$;

c) $\mathsf{bin}(\mathsf{binL}(x), \mathsf{binR}(x)) = x$;

ii.) axioms for the size

a) $\mathsf{size}(\mathcal{E}) = 0$;

b) $\mathsf{size}(\mathsf{bin}(x, y)) = \mathsf{size}(x) + \mathsf{size}(y)$;

iii.) ground literals over the sort TREES

a) $\mathsf{bin}(a, b) = c$;

b) $\mathsf{binL}(d) = e$;

c) $\mathsf{binR}(f) = g$;

d) $h_1 \bowtie h_2$;

iv.) ground literals over the sort AG

a) $\mathsf{size}(\mathcal{E}) = 0$

b) $ns \bowtie m_1t_1 + n_2t_2 + \cdots + m_nt_n$.

where a, b, c, d, e, f, g, h_1 and h_2 are constants of sort TREES, the literals in the set (ivb) are in reductive form, and the summands s, $t_1, \ldots t_n$ are in one of the following shapes: (i) constants of sort AG, (ii) $\mathsf{size}(a)$, (iii) $\mathsf{size}(\mathsf{binL}(b))$, (iv) $\mathsf{size}(\mathsf{binR}(c))$.

Let us now put as ordering a RPO with a total precedence $\succ$ on the symbols of the signature such that all the symbols have a lexicographic status, except $+$, whose status is multiset, and satisfying: (a) $\mathsf{bin} \succ \mathsf{binR} \succ \mathsf{binL} \succ c \succ \mathsf{size}$ for every constant c of sort TREES; (b) $\mathsf{size} \succ g \succ - \succ + \succ 0$ for every constant g of sort AG.

Analogously to what happens in the previous example, these requirements over the precedence guarantee that every compound term of sort TREES is bigger than any constant of the same sort, and that any term of the kind $\mathsf{size}(a)$ is bigger than any constant of sort AG.

Proposition 5. *For any set G of ground literals, any saturation of $T_T \cup T_{size} \cup G$ w.r.t. $\mathcal{SP}_{AG}$ is finite.*

Proof. Let us start with a brief remark. When the calculus $\mathcal{SP}_{AG}$ is applied to terms of sort TREES, it boils down to the standard superposition calculus. So, since any term of sort different from AG cannot contain subterms of sort AG, it is possible to use all the reduction rules of the standard superposition in order to simplify and shorten the saturation. In particular, we will apply in what follows the rule of *simplification*, which allows to substitute to a set of clauses of the kind $S \cup \{C[l']_p, l = r\}$ a set of the kind $S \cup \{C[r\theta]_p, l = r\}$ whenever, for some substitution θ, $l' \equiv l\theta$, $r\theta \prec l\theta$ and, for each literal L in $C[l']_p$, $(l\theta = r\theta) \prec L$. Moreover, we can easily see that the standard rule of *deletion* and *strict subsumption* can be safely applied, without affecting the refutational completeness of $\mathcal{SP}_{AG}$.

Now, if we try to saturate a set of literals of kind as above, we discover that, apart from some tautologies that can be immediately deleted, there are produced five new kind of literals:

- $\mathsf{bin}(z, \mathsf{binR}(\mathsf{bin}(z, t))) = \mathsf{bin}(z, t)$, from (ia) applied to (ic);
- $\mathsf{bin}(\mathsf{binL}(\mathsf{bin}(z, t)), t) = \mathsf{bin}(z, t)$, from (ib) applied to (ic);
- $\mathsf{size}(\mathsf{binR}(t)) = -\mathsf{size}(\mathsf{binL}(t)) + \mathsf{size}(t)$, from (ic) applied to (iib) $(*)$;
- $\mathsf{bin}(e, \mathsf{binR}(d)) = d$, from (iiib) applied to (ic);
- $\mathsf{bin}(\mathsf{binL}(f), g) = f$, from(iiic) applied to (ic).

The literals of the first two kinds are deleted from the saturation because, applying one step of simplification, a tautology is produced (and immediately deleted). When performing any step of saturation involving the literals of the third kind, it is easy to restrict the number of possible inferences observing that, for every grounding and consequent reduction in AG-normal form, the summand that will always be the maximal one is the one obtained from the instantiation of $\mathsf{size}(\mathsf{binR}(t))$. So, the only possible orientations of that kind of literals are exactly the ones in the shape $\mathsf{size}(\mathsf{binR}(t)) = -\mathsf{size}(\mathsf{binL}(t)) + \mathsf{size}(t)$ $(-\mathsf{size}(\mathsf{binR}(t)) = \mathsf{size}(\mathsf{binL}(t)) - \mathsf{size}(t)$ in the inverse rule), and the only allowed splittings in $t_1 + t_2$ are the ones of the kind $t_1 \equiv \mathsf{size}(\mathsf{binR}(t))$ and $t_2 \equiv \mathsf{size}(\mathsf{binL}(t)) - \mathsf{size}(t)$.

At this point it is easy to prove that the saturation is finite, because, when the literal of the kind $(*)$ has been produced, the set of non-ground literals is saturated; moreover, every other literal generated form now on during the saturation phase is ground and it is strictly smaller than the ground literal in the antecedent of the rule used to derive it. Since the ordering on the literals is the multiset extension of a terminating ordering, it is terminating too.

Centre de recherche INRIA Nancy – Grand Est
LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)

Centre de recherche INRIA Bordeaux – Sud Ouest : Domaine Universitaire - 351, cours de la Libération - 33405 Talence Cedex
Centre de recherche INRIA Grenoble – Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier
Centre de recherche INRIA Lille – Nord Europe : Parc Scientifique de la Haute Borne - 40, avenue Halley - 59650 Villeneuve d'Ascq
Centre de recherche INRIA Paris – Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex
Centre de recherche INRIA Rennes – Bretagne Atlantique : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex
Centre de recherche INRIA Saclay – Île-de-France : Parc Orsay Université - ZAC des Vignes : 4, rue Jacques Monod - 91893 Orsay Cedex
Centre de recherche INRIA Sophia Antipolis – Méditerranée : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399