Satisfiability Procedures for Combination of Theories Sharing Integer Offsets*

Enrica Nicolini, Christophe Ringeissen, Michaël Rusinowitch

LORIA & INRIA Nancy Grand Est, France
E-mail: FirstName.LastName@loria.fr

Abstract. We present a novel technique to combine satisfiability procedures for theories that model some data-structures and that share the integer offsets. This procedure extends the Nelson-Oppen approach to a family of non-disjoint theories that have practical interest in verification. The result is derived by showing that the considered theories satisfy the hypotheses of a general result on non-disjoint combination. In particular, the capability of computing logical consequences over the shared signature is ensured in a non trivial way by devising a suitable complete superposition calculus.

1 Introduction

Satisfiability procedures for fragments of Arithmetics and data structures such as arrays and lists are at the core of many state-of-the-art verification tools, and their design and correct implementation is a hard task [5]. To overcome this difficulty, there is an obvious need for developing general and systematic methods to build decision procedures. Two important approaches have been investigated based respectively on combination and rewriting.

The *combination approach* for the satisfiability problem has been initiated in [13,16]. The methodology is to combine existing decision procedures for component theories in order to get a decision procedure for the union of the theories. In particular, the combination à la Nelson-Oppen is the core of many verification tools, even if the implementations often exploit ideas quite far from the original schema (see, e.g. [11,4]). This method assumes that component theories have disjoint signatures. An extension to the non-disjoint case has been proposed in [8,9], where the cooperation between the decision procedures relies on their capabilities of computing logical consequences built over the shared signature.

The *rewriting approach* allows us to flexibly build satisfiability procedures [2,1] based on a general calculus for automated deduction, namely the superposition calculus [15]. Hence, to obtain satisfiability procedures becomes easy by using an (almost) off-the-shelf theorem prover implementing superposition.

* This paper is a presentation-only version of the paper "Satisfiability Procedures for Combination of Theories Sharing Integer Offsets" that appeared in the proceedings of the 15th International Conference on Tools and Algorithms for the Construction and Analysis of Systems, TACAS '09 [14].

These two approaches are complementary for two main reasons. First, combination techniques allow us to incorporate theories which are difficult to handle using rewriting techniques, such as Linear Arithmetics. Second, rewriting techniques are of prime interest to design satisfiability procedures which can be efficiently plugged into the disjoint combination framework [10]. In some particular cases, the rewriting approach is an alternative to the combination approach by allowing us to build superposition-based satisfiability procedures for combinations of finitely axiomatized theories, including the theory of Integer Offsets [1,3], but these theories must be over *disjoint* signatures.

In this paper, we show how to apply a superposition calculus to build decision procedures that can be plugged into the *non-disjoint* combination framework. We focus on theories sharing Integer Offsets. We present a superposition calculus dedicated to this theory and show the soundness of this new calculus for several *non-disjoint* extensions of this theory. The interest of combining counter arithmetic and uninterpreted functions in verification is advocated in [6], where uninterpreted functions are used for abstracting data and Integer Offsets allows us to express counters and a form of pointers, thanks to the successor function s and 0. For instance, the possibility of using Integer Offsets enables us to consider (and combine) several models of lists:

- We can use the classical model of lists à la LISP, using cons, car, cdr operators, augmented with a length function ℓ defined as follows: $\ell(\mathsf{cons}(e, x)) = \mathsf{s}(\ell(x))$ and $\ell(\mathsf{nil}) = 0$. In general, lists are over arbitrary elements but we may use also lists over integer elements.
- We can consider lists defined as records with two fields, the first one for the list itself, and the second one to store its length. Let us consider the operator $\mathsf{rselect_i}$ to access to the i-th field of a record, $rcons(e, r)$ denotes the record obtained by adding an element e to the list of r, and $rnil$ denotes the record corresponding to the empty list, we have the following axiomatization:

$$\begin{array}{ll} \mathsf{rselect}_1(rcons(e, r)) = \mathsf{cons}(e, \mathsf{rselect}_1(r)) & \mathsf{rselect}_1(rnil) = nil \\ \mathsf{rselect}_2(rcons(e, r)) = \mathsf{s}(\mathsf{rselect}_2(r)) & \mathsf{rselect}_2(rnil) = 0 \end{array}$$

  This model of lists can be seen as a refinement of the first model in which one has a direct access to its “cardinality”.

The combination framework presented in the paper can be applied to decide the satisfiability of ground formulas expressed in the union of these two models of lists (provided both models use distinct names for list operators). Roughly speaking, such combination is useful to verify for instance that two programs written using different models of lists are “equivalent”.

Plan of the paper. After this introduction, Section 2 gives the main concepts and notations related to first-order theories. Section 3 recalls the non-disjoint combination framework of [9]. In Section 4, we present a superposition calculus dedicated to the theory of Integer Offsets. In Section 5, we give some examples of theories for which this superposition calculus can be turned into decision procedures. In Section 6, we show that this superposition calculus can be also applied to deduce logical shared consequences. Moreover, all the requirements for applying the non-disjoint combination framework of [9] are satisfied by the extensions of Integer Offsets we are interested in. Finally, Section 7 concludes with some final remarks and some hints of future work. For lack of space, proofs are omitted and can be found in [14].

2 Preliminaries

A *signature* Σ is a set of functions and predicate symbols (each endowed with the corresponding arity). We assume the binary equality predicate symbol '$=$' to be always present in any signature Σ (so, if $\Sigma = \emptyset$, then Σ does not contain other symbols than equality). The signature obtained from Σ by adding a set $\underline{a}$ of new constants (i.e., 0-ary function symbols) is denoted by $\Sigma^{\underline{a}}$. *Σ-atoms*, *Σ-literals*, *Σ-clauses*, and *Σ-formulae* are defined in the usual way. 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. Given a function symbol f, a f-rooted term is a term whose top-symbol is f.

From the semantic side, we have the standard notion of a *Σ-structure* $\mathcal{M} = (M, \mathcal{I})$: this is a support set M endowed with an 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 function and predicate symbols from $\Sigma \setminus \Sigma_0$.

A collection of Σ-sentences is a Σ-theory, and a Σ-theory T admits *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, also called the T-satisfiability problem, which is the problem of deciding whether a Σ-constraint is satisfiable in a model of T (and, if so, we say that the constraint is T-satisfiable). 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.

3 Background on Combining Theories

We are interested in applying the general method developed in [9] for the combination of satisfiability procedures in unions of non-disjoint theories. This method extends the Nelson-Oppen combination schema known for unions of signature-disjoint theories, and leads to the following result:

Theorem 1. *[9] Consider two theories T_1, T_2 in signatures Σ_1, Σ_2 and suppose that:*

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

The decidability result of Theorem 1 is obtained by relying on the available decision procedures for T_1 and T_2, and cooperating them through an exchange of information over the shared signature $\Sigma_1 \cup \Sigma_2$. There are three crucial points in this schema: first of all, one should identify conditions sufficient to guarantee the correctness of the resulting procedure: this has been addressed requiring that the component theories must be both "compatible" with respect to a common sub-theory. Secondly, one should ensure the capability of computing the information to be exchanged: this issue is encoded into the requirement that the two theories T_1 and T_2 are "effectively Noetherian extensions" of a common sub-theory T_0. Finally, one should guarantee that the exchange process eventually halts: the termination of the whole procedure is ensured thanks to the so called Noetherianity of T_0. Let us explain in more details what the aforementioned requirements are.

Definition 1 (T_0-compatibility [8]). *Let T be a theory in the signature Σ and let T_0 be a universal theory in a subsignature $\Sigma_0 \subseteq \Sigma$. We say that T is* T_0-compatible *iff $T_0 \subseteq T$ and there is a Σ_0-theory T_0^* 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 $T_0^\star$;
(iv) every Σ-constraint which is satisfiable in a model of T is satisfiable also in a model of $T_0^\star \cup T$.

The requirements (i) to (iii) make the theory $T_0^\star$ unique, provided it exists ($T_0^\star$ is the so-called *model completion* of T_0). These requirements are a generalization of the stable infiniteness requirement of the Nelson-Oppen combination procedure: in fact, if T_0 is the empty theory in the empty signature, $T_0^\star$ is the theory axiomatizing an infinite domain, so that (iii) holds trivially and (iv) is precisely stable infiniteness.

Example 1. Let us consider the theory of Integer Offsets T_I:

$\boxed{T_I}$ rules the behaviour of the successor function s and the constant 0. T_I has the mono-sorted signature $\Sigma_I := \{0 : \text{INT}, \mathsf{s} : \text{INT} \to \text{INT}\}$, and it is axiomatized as follows:

$\forall x\ \mathsf{s}(x) \neq 0$
$\forall x, y\ \mathsf{s}(x) = \mathsf{s}(y) \to x = y$
$\forall x\ x \neq t(x)$ for all the terms $t(x)$ over Σ_I that properly contain x

T_I is a universal theory that admits model completion: indeed, if we add to T_I the axiom $\forall x(x \neq 0 \to \exists y\, x = \mathsf{s}(y))$, we obtain a theory $T_I^\star$ that admits quantifier elimination (see, e.g. [7]) and such that every constraint that is satisfiable in a model of T_I is satisfiable also in a model of $T_I^\star$. To justify the last claim, it is sufficient to observe that each model of T_I can be extended to a model of $T_I^\star$ simply by adding recursively to each element different from (the interpretation of) 0 a "predecessor". Since this operation does not affect the truth of any constraint, we obtain that the condition (iii) is satisfied.

Now, for any theory $T \supseteq T_I$ over a signature $\Sigma \supseteq \Sigma_I$ the T_I-compatibility requirement simply reduces to the following condition: every constraint Γ that is satisfiable in a model of T must be satisfiable also in a model of $T \cup \forall x(x \neq 0 \to \exists y\, x = \mathsf{s}(y))$.

The method for cooperating the satisfiability procedures makes use of the capability of deducing logical consequences over the shared signature. In order to ensure the termination when deducing those logical consequences, we rely on Noetherian theories. Intuitively, a theory is Noetherian if there exists only a finite number of atoms that are not redundant when reasoning modulo T_0.

Definition 2 (Noetherian Theory [9]). *A Σ_0-theory T_0 is* Noetherian *if and only if 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 an n such that $T_0 \cup \Theta_n \models A$, for every natural number m and atom $A \in \Theta_m$.

Example 2. (Example 1 continued). Many examples of Noetherian theories come from the formalization of algebraic structures, but an interesting class of Noetherian theories consists in all the theories whose signature contains only constants and one unary function symbol [17]. Thus, the theory of Integer Offsets T_I enjoys this property.

Let us consider now a theory $T \supseteq T_0$ with signature $\Sigma \supseteq \Sigma_0$, and suppose we want to discover, given an arbitrary set of ground clauses Θ over Σ, a "complete set" of logical positive consequences of Θ over Σ_0, formalized by the notion of T_0*-basis.*

Definition 3 (T_0-basis). *Given a finite set Θ of ground clauses (built out of symbols from Σ and possibly further free constants) and a finite set of free constants $\underline{a}$, a T_0*-basis *for Θ w.r.t. $\underline{a}$ is a set Δ of* positive *ground $\Sigma_0^{\underline{a}}$-clauses such that*

(i) $T \cup \Theta \models C$, for all $C \in \Delta$ and
(ii) if $T \cup \Theta \models C$ then $T_0 \cup \Delta \models C$, for every positive ground $\Sigma_0^{\underline{a}}$-clause C.

Notice that in the definition of a basis we are interested only in positive ground clauses: the exchange of positive information is sufficient to ensure the completeness of the resulting procedure. The interest in Noetherian theories lies in the fact that, for every set of Σ-clauses Θ and for every finite set $\underline{a}$ of constants, a finite T_0-basis for Θ w.r.t. $\underline{a}$ always exists (Proposition 3.22 in [9]). Unfortunately, a basis for a Noetherian theory needs not to be computable; this motivates the following definition corresponding to the last hypothesis of Theorem 1:

Definition 4 ([9]). *Given a finite set* $\underline{a}$ *of free constants, a* T-residue enumerator for T_0 w.r.t. $\underline{a}$ *is a computable function* $Res^{\underline{a}}_{T}(\Gamma)$ *mapping a set of* Σ*-clauses* Γ *to a finite* T_0*-basis for* Γ *w.r.t.* $\underline{a}$[1]. *A theory* T *is an* effectively Noetherian extension of T_0 *if and only if* T_0 *is Noetherian and there exists a* T*-residue enumerator for* T_0 *w.r.t. every finite set* $\underline{a}$ *of free constants.*

Now we are ready to give a more detailed picture of the procedure that is the core of Theorem 1, and that extends the Nelson-Oppen combination method to theories over non disjoint signatures.

Algorithm 1 Extending Nelson-Oppen

```
Step 1. Purify the finite input set of ground $(\Sigma_1 \cup \Sigma_2)^{\underline{b}}$-literals $\Gamma$, thus producing a
    finite set $\Gamma_1$ of ground $\Sigma_1^{\underline{a}}$-literals and finite set $\Gamma_2$ of ground $\Sigma_2^{\underline{a}}$-literals s.t. $\Gamma_1 \cup \Gamma_2$
    is $T_1 \cup T_2$-equisatisfiable with $\Gamma$.
Step 2. Using the $T_i$-residue enumerator $Res^{\underline{a}}_{T_i}$, check the output of $Res^{\underline{a}}_{T_i}(\Gamma_i)$:
    If $Res^{\underline{a}}_{T_i}(\Gamma_i) = \Delta_i$ and $\Delta_i \neq \bot$ for each $i \in \{1, 2\}$, then
        Step 2.1. For each $D \in \Delta_i$ such that $T_j \cup \Gamma_j \not\models D$, $(i \neq j)$, add $D$ to $\Gamma_j$
        Step 2.2. If $\Gamma_1$ or $\Gamma_2$ has been changed in Step 2.1, then rerun Step 2
    Else return "unsatisfiable"
Step 3. If this step is reached, return "satisfiable".
```

In the following we will show how to discover theories that are amenable to be combined via the above schema and that share the theory of Integer Offsets. More in detail, we will focus on a particular extension of the superposition calculus that will proved to be a decision procedure for theories extending T_I and that will provide residue enumerators for T_I.

4 Superposition Calculus for Integer Offsets

Recent literature has focused on the possibility of using the superposition calculus in order to decide the satisfiability of ground formulae modulo the theory of Integer Offsets and some disjoint extensions [1,3]. Contrary to those papers, we are interested in a superposition-based calculus to deal with non-disjoint extensions of Integer Offsets, being able to constraint the successor symbol with additional axioms.

[1] If Γ is T-unsatisfiable, then without loss of generality a residue enumerator can always return the singleton set containing the empty clause.

Let us consider the axiomatization of the theory of Integer Offsets T_I defined in Example 1. Our aim is to develop a calculus able to take into account the axioms of T_I into a framework based on superposition. To this aim, let us consider a presentation of the superposition calculus specialized for reasoning over sets of literals, whose rules are described in Figures 1 and 2, augmented with the four rules over ground terms presented in Figure 3. as usual, we assume a term reduction ordering $\prec$ which is total on ground terms.

Superposition	$\dfrac{l[u'] = r \quad u = t}{(l[t] = r)\sigma}$	$(i), (ii)$
Paramodulation	$\dfrac{l[u'] \neq r \quad u = t}{(l[t] \neq r)\sigma}$	$(i), (ii)$
Reflection	$\dfrac{u' \neq u}{\bot}$	

where σ is the most general unifier of u and u', u' is not a variable in *Superposition* and *Paramodulation*, L is a literal, $\bot$ is the syntactic sign used to denote the inconsistency and the following hold: *(i)* $u\sigma \not\preceq t\sigma$, *(ii)* $l[u']\sigma \not\preceq r\sigma$.

Fig. 1. Expansion Inference Rules.

Let us adapt the standard definition of *derivation* to the calculus we are interested in:

Definition 5. *Let $\mathcal{SP}_I$ be the calculus depicted in Figures 1, 2 and 3. A derivation (δ) with respect to $\mathcal{SP}_I$ is a (finite or infinite) sequence of sets of literals $S_1, S_2, S_3, \ldots, S_i, \ldots$ such that, for every i, it happens that:*

(i) S_{i+1} is obtained from S_i adding a literal obtained by the application of one of the rules in Figures 1, 2 and 3 to some literals in S_i;
(ii) S_{i+1} is obtained from S_i removing a literal according to one of the rules in Figures 2 or to the rule R1 or R2.

If we focus on the rules of Simplification, R1 and R2, we notice that the effects of the application of any of these rules involve two steps in the derivation: in the former a new literal is added, and in the latter a literal is deleted.

If S is a set of literals, let GS be the set of all the ground instances of S. A literal L is said to be *redundant* with respect to a set of literals S if, for all the ground instances $L\sigma$ of L, it happens that $\{E \mid E \in GS \ \&\ E \prec \ L\sigma\} \models L\sigma$. We notice that in our derivations only redundant literals are deleted:

Fact. If in a derivation S_{i+1} is equal to $S_i \setminus \{L\}$, then L is redundant with respect to S_i.

Proof. The claim above is well known if S_{i+1} is obtained from S_i applying one of the rules in Figure 2, and it follows immediately in the case we are applying R1 or R2.

Subsumption	$\dfrac{S \cup \{L, L'\}}{S \cup \{L\}}$	if $L\vartheta \equiv L'$ for some substitution ϑ
Simplification	$\dfrac{S \cup \{L[l'], l = r\}}{S \cup \{L[r\vartheta], l = r\}}$	if $l' \equiv l\vartheta$, $r\vartheta \prec l\vartheta$, and $(l\vartheta = r\vartheta) \prec L[l\vartheta]$
Deletion	$\dfrac{S \cup \{t = t\}}{S}$	

where L and L' are literals and S is a set of literals.

Fig. 2. Contraction Inference Rules.

R1	$\dfrac{S \cup \{\mathsf{s}(u) = \mathsf{s}(v)\}}{S \cup \{u = v\}}$	if u and v are ground terms
R2	$\dfrac{S \cup \{\mathsf{s}(u) = t, \mathsf{s}(v) = t\}}{S \cup \{\mathsf{s}(v) = t, u = v\}}$	if u, v and t are ground terms and $\mathsf{s}(u) \succ t$, $\mathsf{s}(v) \succ t$ and $u \succ v$
C1	$\dfrac{S \cup \{\mathsf{s}(t) = 0\}}{S \cup \{\mathsf{s}(t) = 0\} \cup \bot}$	if t is a ground term
C2	$\dfrac{S \cup \{\mathsf{s}^n(t) = t\}}{S \cup \{\mathsf{s}^n(t) = t\} \cup \bot}$	if t is a ground term and $n \in \mathbb{N}$

where S is a set of literals and $\bot$ is the symbol for the inconsistency.

Fig. 3. Ground Reduction Inference Rules.

So, as usual, we label with S_∞ the set of literals generated during a derivation δ (in symbols, $S_\infty = \bigcup_i S_i$), and with S_ω the set of persistent literals of δ: $S_\omega = \bigcup_i \bigcap_{j>i} S_j$. We adopt the standard definition for a rule π of the calculus being *redundant* with respect to a set of clauses S whenever, for every ground instance of the rule $\pi\sigma$ it happens that $\{E \mid E \in GS \ \& \ E \prec \ C_m\sigma\} \models D\sigma$, where $C_m\sigma$ is the maximal clause in the antecedent, and $D\sigma$ is the consequent of the rule. According to this definition, a derivation w.r.t. $\mathcal{SP}_I$ is *fair* if, for every literal $L_1, L_2, \ldots, L_m \in S_\omega$, every rule that has $L_1, \ldots, L_m$ as premises is redundant w.r.t. S_∞.

Suppose now to take into account a fair derivation δ. We notice that, if a literal L is added at a certain step of the derivation, say S_{i+1}, then L is either a logical consequence of some literals in S_i, or it is a consequence of some literals in S_i and the axioms of the theory T_I. Thus:

Proposition 1. *If the set of persistent literals S_ω contains $\bot$, then S_ω is unsatisfiable in any model of T_I.*

On the other hand, since the reduction rules we can apply during the derivation satisfy the general requirements about the redundancy, we have that:

Proposition 2. *If the set of persistent literals S_ω does not contain $\bot$, then S_ω is satisfiable.*

What remains to show is that this calculus is *refutationally complete* with respect to the models of T_I (namely the structures in which the function s is injective, acyclic and such that 0 does not belong to the image of s). We want to identify in the following at least one case in which the calculus in Figures 1, 2 and 3 is not only refutationally complete w.r.t. T_I, but it is complete, too.

Remark 1. Since the satisfiability of S_ω is equivalent to the satisfiability of S_∞, and since the satisfiability of each step S_{i+1} in the derivation implies the satisfiability of S_i, we have in particular that if S_ω is satisfiable, then S_0 is satisfiable. Moreover, it is immediate to check that the unsatisfiability in the models of T_I of S_ω implies the unsatisfiability of S_0 in the same class of structures. So, in case it happens that the calculus described in Figures 1, 2 and 3 is complete, we can proceed as usual when considering procedures based on saturation methods: an initial set of literals S_0 will be satisfiable (in a model of T_I) if and only if its saturation S_ω does not contain $\bot$.

4.1 Completeness

From now on, we assume that the ordering we consider when performing any application of $\mathcal{SP}_I$ is T_I-*good*:

Definition 6. *We say that an ordering $\succ$ over terms on a signature containing Σ_I is T_I-*good *whenever it satisfies the following requirements:*

(i) *$\succ$ is a simplification ordering that is total on ground terms;*
(ii) *0 is minimal;*
(iii) *whenever two terms t_1 and t_2 are not s-rooted it happens that $\mathsf{s}^{n_1}(t_1) \succ \mathsf{s}^{n_2}(t_2)$ iff either $t_1 \succ t_2$ or ($t_1 \equiv t_2$ and n_1 is bigger than n_2).*

Proposition 3. *Assuming T_I-*good *ordering $\succ$ over terms, if the set of persistent literals S_ω satisfies the following assumptions:*

- *S_ω does not contain $\bot$,*
- *S_ω does not contain equation whose maximal term is a variable of sort* INT*, and s-rooted terms can be maximal just in ground equations.*

then S_ω is satisfiable in a model of T_I.

Collecting all the results obtained so far, we can conclude that:

Theorem 2. *Let T be a Σ-theory presented as a finite set of unit clauses such that $\Sigma \supseteq \Sigma_I$, and assume to put an ordering over terms that is T_I-*good*. $\mathcal{SP}_I$ induces a decision procedure for the constraint satisfiability problem w.r.t. $T \cup T_I$ if, for any set G of ground literals:*

- *the saturation of $Ax(T) \cup G$ w.r.t. $\mathcal{SP}_I$ is finite,*
- *the saturation of $Ax(T) \cup G$ w.r.t. $\mathcal{SP}_I$ does not contain non-ground equations whose maximal term is s-rooted, or equations whose maximal term is a variable of sort* INT.

4.2 Termination

Proposition 4. *For any set G of ground literals over a signature extending Σ_I, any saturation of G w.r.t. $\mathcal{SP}_I$ is finite.*

Proof. Each step either adds a literal that is smaller than (at least) one literal already present in the saturation, or delete one literal, hence the multiset of literals decreases according to the well-founded ordering $((\succ)^{mul})^{mul}$.

Corollary 1. *$\mathcal{SP}_I$ induces a decision procedure for the constraint satisfiability problem w.r.t. the union of T_I and the theory of equality.*

5 Examples of Integer Offsets Extensions

We investigate theories sharing symbols of T_I in a specific way, thanks to axioms of the form $g(f(\dots, x, \dots)) = \mathsf{s}(g(x))$ where f, g are function symbols not occurring in Σ_I. Despite this restricted form of axioms, we are already able to consider interesting examples of Integer Offsets extensions.

5.1 Lists with Length

Let us consider T_{LLI}, the theory of lists endowed with length. T_{LLI} can be axiomatized as the union of the theories T_L, T_ℓ and T_I, where T_I is the theory of Integer Offsets of Example 1 and:[2]

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

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

$\boxed{T_\ell}$ is the theory that gives the axioms for the function length $\ell : \textsc{lists} \to \textsc{int}$:

$$\begin{gathered} \ell(\mathsf{nil}) = 0 \\ \ell(\mathsf{cons}(x, y)) = \mathsf{s}(\ell(y)) \end{gathered}$$

We want to show that the constraint satisfiability problem for T_{LLI} is decidable via the calculus described in the previous section.

[2] All the axioms should be considered as universally quantified.

First: reduction We start addressing the problem of checking the satisfiability of a constraint w.r.t. T_{LLI}. Let G be a set of ground literals over $\Sigma_{T_{LLI}}$; we can associate to G the set of formulae G' obtained by replacing 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 (this is the same reduction used in [2]).

Let now $T_{L'}$ be the subtheory of T_L whose axioms are just the (equational) axioms in the left column of the presentation of T_L. We have that:

Proposition 5. *G is satisfiable w.r.t. T_{LLI} if and only if G' is satisfiable w.r.t. $T_{L'} \cup T_\ell \cup T_I$.*

Second: saturation According to Proposition 5 and applying at most some standard steps of flattening, we can focus our attention to sets of literals of the following kinds (x is a variable of sort ELEM, y is a variable of sort LISTS, $h, l, a, f, g, l_1, l_2, e, d, e_1, e_2, i, i_1, i_2$ are constants of the appropriate sorts and the symbol $\bowtie$ is a shortening for both $=$ and $\neq$), 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 $\neg\mathsf{atom}$
  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) $\ell(\mathsf{cons}(x, y)) = \mathsf{s}(\ell(y))$;
iv.) ground literals over the sort LISTS
  a) $\mathsf{cons}(e, l) = c$;
  b) $\mathsf{cdr}(f) = g$;
  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 INT
  a) $\ell(a) = \mathsf{s}^m(i)$;
  b) $\mathsf{s}^m(i_1) \neq \mathsf{s}^n(i_2)$;
  c) $\mathsf{s}^n(i_1) = i_2$;
  d) $i_1 = \mathsf{s}^n(i_2)$.

Let us choose, as ordering over the terms, a LPO ordering $\succ$ whose underlying precedence over the symbols of the signature respects the following requirements:

- $\mathsf{cons} > \mathsf{cdr} > \mathsf{car} > c > e > \ell$ for every constant c of sort LISTS and every constant e of sort ELEM;
- $\ell > i > 0 > \mathsf{s}$ for every constant i of sort INT;

These requirements over the precedence guarantee that every compound term of sort LISTS is bigger than any constant, any compound term over the sort ELEM is bigger than any constant, and that $\succ$ is a T_I*-good* ordering.

We require that the rules in Figures 2 and 3 are applied, whenever possible, before the rules in Figure 1 (in other words we require that the contraction rules have a higher priority).

Proposition 6. *For any set G of ground literals, any saturation of $Ax(T_{LLI}) \cup G$ w.r.t. $\mathcal{SP}_I$ is finite.*

The key observations, in order to prove termination, are that the non-ground set of literals is already saturated, every 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.

Moreover, since in the saturation no non-ground equation whose maximal term is s-rooted is generated, we can conclude by Theorem 2 that $\mathcal{SP}_I$ is a decision procedure for the constraint satisfiability problem w.r.t. T_{LLI}.

5.2 Lists over Integer Elements

Let us consider now lists whose elements are integers. The reduction of Section 5.1 works without any changes, so we can check if the calculus developed in Figures 1, 2 and 3 is still a decision procedure for the constraint satisfiability problem of lists with length and integer elements. We can apply at most some standard steps of flattening and we focus our attention to sets of literals of the kinds i—iv) defined in Section 5.1 plus the new following one which merges the kinds v—vi) of Section 5.1:

v.) ground literals over the sort INT

a) $\mathsf{car}(h) = \mathsf{s}^n(i)$;
b) $\ell(a) = \mathsf{s}^m(i)$;
c) $\mathsf{s}^m(i_1) \neq \mathsf{s}^n(i_2)$;
d) $\mathsf{s}^n(i_1) = i_2$;
e) $i_1 = \mathsf{s}^n(i_2)$.

Let us put over the symbols of the signature an order that respects the same requirements we have asked in Section 5.1. The same remarks about termination and the shape of the saturated set of the previous section apply also to this case, guaranteeing that $\mathcal{SP}_I$ provides a decision procedure.

5.3 Records with Increment

Let us consider records in which all the attribute identifiers are associated to the same sort INT, and suppose we want to be able to increment by a unity every value stored into the record. To formalize this situation, we can choose a signature as follows: let $Id = \{id_1, id_2, \dots, id_n\}$ a set of attribute identifiers and let us name REC the sort of records; for every attribute identifier $id_1, id_2, \dots, id_n$ we have a couple of functions $\mathsf{rselect_i} : \text{REC} \to \text{INT}$ and $\mathsf{rstore_i} : \text{REC} \times \text{INT} \to \text{REC}$; moreover, there is also the increment function $\mathsf{incr} : \text{REC} \to \text{REC}$. The axioms of the theory of integer records with increment, T_{IRI}, are the following:

$\boxed{T_{IRI}}$: for every i, j such that $1 \leq i, j \leq n$, $i \neq j$

$$
\begin{aligned}
\mathsf{rselect_i}(\mathsf{rstore_i}(x, y)) &= y \\
\mathsf{rselect_j}(\mathsf{rstore_i}(x, y)) &= \mathsf{rselect_j}(x) \\
\wedge_{i=1}^{n}(\mathsf{rselect_i}(x) = \mathsf{rselect_j}(y)) &\rightarrow x = y \qquad \text{(extensionality)} \\
\mathsf{rselect_i}(\mathsf{incr}(x)) &= \mathsf{s}(\mathsf{rselect_i}(x))
\end{aligned}
$$

In order to check the satisfiability of a set of ground literals w.r.t. T_{IRI}, we notice that every literal of the kind $r_1 \neq r_2$ is equivalent to a clause of the kind $\bigvee_{i=1}^{n} \mathsf{rselect_i}(r_1) \neq \mathsf{rselect_i}(r_2)$, so can we substitute every disequation between records with the corresponding clause and then check the satisfiability of the resulting set of clauses by case split.

So we can restrict our attention to sets of literals in which no disequation between records appears. In this case, following the same argument used in [1], it is possible to check the satisfiability forgetting the extensionality axioms (the presence of the function incr does not affect the argument). Thus we are reduced to consider the saturation of sets of literals of the following kind:

i.) equational axioms for records
  a) $\mathsf{rselect_i}(\mathsf{rstore_i}(x, y)) = y$;
  b) $\mathsf{rselect_j}(\mathsf{rstore_i}(x,y)) = \mathsf{rselect_j}(x)$;
  c) $\mathsf{rselect_i}(\mathsf{incr}(x)) = \mathsf{s}(\mathsf{rselect_i}(x))$;
ii.) ground literals over the sort REC
  a) $r_1 = r_2$;
  b) $\mathsf{rstore_i}(r_1, \mathsf{s}^n(k)) = r_2$;
  c) $\mathsf{incr}(r_1) = r_2$;
iii.) ground literals over the sort INT
  a) $\mathsf{rselect_i}(r) = \mathsf{s}^n(k)$;
  b) $\mathsf{s}^n(k_1) = k_2$;
  c) $k_1 = \mathsf{s}^n(k_2)$;
  d) $\mathsf{s}^n(k_1) \neq \mathsf{s}^m(k_2)$.

where x is a variable of sort REC, y is a variable of sort INT, and r, r_1, r_2, k, k_1, k_2 are constants of appropriate sorts. As usual, let us consider a LPO ordering over terms such that the underlying precedence over the symbols in the signature satisfies the following requirements: for all i, j in $\{1, \ldots, n\}$, $\mathsf{incr} > \mathsf{rstore_i}$, $\mathsf{rstore_i} > \mathsf{rselect_j}$, $\mathsf{rselect_i} > c$ for every constant c and every constant c is such that $c > 0 > \mathsf{s}$.

Proposition 7. *For any set G of ground literals, any saturation of $Ax(T_{IRI}) \cup G$ w.r.t. $\mathcal{SP}_I$ is finite.*

The completeness of the calculus can be shown relying on the observation that no non-ground literals involving the function symbol s are generated, and that the chosen ordering is a T_I-*good* one.

6 Combination of Theories Sharing Integer Offsets

In the previous section we have collected examples of theories extending the theories of the Integers Offsets T_I and whose constraint satisfiability problem is decidable. We have already noticed that T_I admits a model completion $T_I^\star$ and that it is a Noetherian theory; to guarantee that these theories can be combined together it is sufficient to show that they satisfy the requirement of being T_I-compatible and effectively Noetherian extensions of T_I.

6.1 T_I-Compatibility

Being for a theory $T \supseteq T_I$ a T_I-compatible theory means that every constraint that is satisfiable w.r.t. T is satisfiable also in a model in which the axiom $\forall x(x \neq 0 \rightarrow \exists y\, x = \mathsf{s}(y))$ holds. To see that actually it is the case for all the theories considered in Section 5, it is sufficient to check that any model of that theories can always be extended, if needed, adding recursively to each element that is different from (the interpretation of) 0 its predecessor and, in case it is needed, modifying accordingly the remaining part of the structure; and to check that this enlargement does not affect the validity both of the constraints that are verified in the structure and of the axioms of the theory. For example, we consider in the Appendix the case of the theory of lists over integer elements with length. Using similar (or simpler) arguments as the ones for this case, it is possible to verify that all the theories in Section 5 are T_I-compatible.

6.2 Derivation of T_I-bases

We have considered Horn Σ'-theories $T' = T \cup T_I$ extending T_I with some theories T axiomatized by unit clauses and we have shown under which assumptions the Superposition Calculus $\mathcal{SP}_I$ is complete w.r.t. T'-satisfiability problems. Let us show that $\mathcal{SP}_I$ allows us to derive T_I-basis. Assume that $G(\underline{a}, \underline{b})$ is a set of ground literals over an expansion of Σ' with the finite sets of fresh constants $\underline{a}, \underline{b}$. Our claim is the following: if S_ω is the saturation of $Ax(T) \cup G(\underline{a}, \underline{b})$ and assuming a T_I-*good* order over the terms in the signature $\Sigma' \cup \{\underline{a}, \underline{b}\}$ such that every term over the subsignature $\Sigma_I^{\underline{a}}$ is smaller than any term that contains a symbol in $(\Sigma' \setminus \Sigma_I) \cup \{\underline{b}\}$, then the subset of OGS_ω over the signature $\Sigma_I^{\underline{a}}$, denoted by $\Delta(\underline{a})$, is a T_I-basis. Since T' is a Horn theory, it is convex and so we can focus our attention just over equations instead of positive (ground) clauses.

Proposition 8. *If $l = r$ is an equation over $\Sigma_I^{\underline{a}}$ implied by $T' \cup G(\underline{a}, \underline{b})$, then $l = r$ is already implied by $T_I \cup \Delta(\underline{a})$, whenever S_ω is (i) finite, (ii) does not $\perp$, and such that (iii) s-rooted terms can be maximal just in ground equations in S_ω and (iv) variables of sort* INT *are never the maximal term into the equations.*

7 Conclusion

We have shown how to apply a superposition calculus to build decision procedures for some theories sharing Integer Offsets. These theories and the related decision procedures satisfy all the requirements for their applications in a non-disjoint combination framework. To the best of our knowledge, this paper is the first contribution showing the interest of a superposition calculus for non-disjoint combinations. This paper paves the way of using non-disjoint combinations (with a shared fragment of Arithmetics) in the context of verification. There are several research directions we want to investigate. Currently, the soundness of the superposition calculus is proved manually for each theory considered in the paper. It would be very interesting to have an automatic proof mechanism using for instance a meta-saturation calculus [12]. Moreover, the considered fragment of Arithmetics is not very expressive and we have some limitations on the form of axioms we are able to handle. Further works are needed to go beyond these restrictions.

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).
2. A. Armando, S. Ranise, and M. Rusinowitch. A rewriting approach to satisfiability procedures. *Information and Computation*, 183(2):140–164, 2003.
3. M. P. Bonacina and M. Echenim. On variable-inactivity and polynomial T-satisfiability procedures. *Journal of Logic and Computation*, 18(1):77–96, 2008.
4. M. Bozzano, R. Bruttomesso, A. Cimatti, T. A. Junttila, S. Ranise, P. van Rossum, and R. Sebastiani. Efficient theory combination via boolean search. *Information and Computation*, 204(10):1493–1525, 2006.
5. A. Bradley and Z. Manna. *The Calculus of Computation*. Springer, 2007.
6. R. E. Bryant, S. K. Lahiri, and S. A. Seshia. Modeling and verifying systems using a logic of counter arithmetic with lambda expressions and uninterpreted functions. In *Proc. of CAV 2002*, volume 2404 of *LNCS*, pages 78–92, Copenhagen (Denmark), 2002. Springer-Verlag.
7. H. B. Enderton. *A Mathematical Introduction to Logic*. Academic Press, New York-London, 1972.
8. S. Ghilardi. Model theoretic methods in combined constraint satisfiability. *Journal of Automated Reasoning*, 33(3-4):221–249, 2004.
9. S. Ghilardi, E. Nicolini, and D. Zucchelli. A comprehensive combination framework. *ACM Transactions on Computational Logic*, 9(2):1–54, 2008.
10. H. Kirchner, S. Ranise, C. Ringeissen, and D.-K. Tran. On superposition-based satisfiability procedures and their combination. In *Proc. of ICTAC 2005*, volume 3722 of *LNCS*, pages 594–608, Hanoi (Vietnam), 2005. Springer-Verlag.
11. S. Krstić, A. Goel, J. Grundy, and C. Tinelli. Combined satisfiability modulo parametric theories. In *Proc. of TACAS 2007*, volume 4424 of *LNCS*, pages 618–631, Braga (Portugal), 2007. Springer.
12. C. Lynch and D.-K. Tran. Automatic decidability and combinability revisited. In *Proc. of CADE-21*, volume 4603 of *LNCS*, pages 328–344, Bremen (Germany), 2007. Springer.
13. G. Nelson and D. C. Oppen. Simplification by cooperating decision procedures. *ACM Transaction on Programming Languages and Systems*, 1(2):245–257, 1979.
14. 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. Extended version as INRIA Report RR-6697.
15. 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.
16. R. E. Shostak. Deciding combinations of theories. *J. of the ACM*, 31:1–12, 1984.
17. D. Zucchelli. *Combination Methods for Software Verification*. PhD thesis, Università degli Studi di Milano and Université Henri Poincaré - Nancy 1, 2008.