Data Structures with Arithmetic Constraints: a Non-Disjoint Combination

Enrica Nicolini, Christophe Ringeissen, Michael Rusinowitch

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Data Structures with Arithmetic Constraints: a Non-Disjoint Combination

Enrica Nicolini — Christophe Ringeissen — Michaël Rusinowitch

N° 6963

Juin 2009

Thème SYM

ISSN 0249-6399 ISRN INRIA/RR--6963--FR+ENG

INSTITUT NATIONAL
DE RECHERCHE
EN INFORMATIQUE
ET EN AUTOMATIQUE

INRIA

centre de recherche
NANCY - GRAND EST

Data Structures with Arithmetic Constraints: a Non-Disjoint Combination

Enrica Nicolini*, Christophe Ringeissen* , Michaël Rusinowitch*

Thème SYM — Systèmes symboliques
Équipe-Projet Cassis

Rapport de recherche n° 6963 — Juin 2009 — 23 pages

Abstract: We apply an extension of the Nelson-Oppen combination method to develop a decision procedure for the non-disjoint union of theories modeling data structures with a counting operator and fragments of arithmetic. We present some data structures and some fragments of arithmetic for which the combination method is complete and effective. To achieve effectiveness, the combination method relies on particular procedures to compute sets that are representative of all the consequences over the shared theory. We show how to compute these sets by using a superposition calculus for the theories of the considered data structures and various solving and reduction techniques for the fragments of arithmetic we are interested in, including Gauss elimination, Fourier-Motzkin elimination and Groebner bases computation.

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

* 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

Structures de données avec contraintes arithmétiques: une combinaison non-disjointe

Résumé :

Nous appliquons une extension de la méthode de combinaison de Nelson-Oppen pour développer une procédure de décision pour un mélange non-disjoint de théories modélisant des structures de données avec un opérateur de comptage et des fragments arithmétiques. Nous présentons des structures de données et des fragments arithmétiques pour lesquelles la méthode de combinaison est complète et effective. Pour être effective, la procédure de combinaison utilise des procédures spécifiques permettant de calculer une représentation de l'ensemble des conséquences logiques exprimées sur la signature partagée d'une formule satisfiable. Nous montrons comment construire de telles procédures en utilisant un calcul de superposition basé sur des techniques de réécriture pour les structures de données et des méthodes de résolution classiques pour l'arithmétique comme l'élimination de Gauss, l'élimination de Fourier-Motzkin et le calcul de bases de Groebner.

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

1 Introduction

Many verification problems can be reduced to checking the satisfiability of formulae modulo a combination of theories including arithmetic operators, the theory of equality, and sophisticated data structures such as lists, arrays, trees, etc. Even uninterpreted function symbols (in other words, the theory of equality) can be used as a possible abstraction to store elements. The classical Nelson-Oppen combination method [14] allows us to combine this very basic data structure with fragments of arithmetic, and the arithmetic with uninterpreted function symbols represents a popular case study due to its practical interest for verification. But the Nelson-Oppen method can be used also to combine the arithmetic with more sophisticated data structures. For instance, the theory of lists and the theory of arrays were already discussed in the seminal paper by Nelson-Oppen [14]. More recently, the development of combination methods and decision procedures has received a lot of interest for its application to solvers for the problem of Satisfiability Modulo Theories (SMT). To improve the applicability of SMT solvers, it is important to develop general uniform methods to combine and to build decision procedures. Hence, equational theorem proving has been successfully applied to build decision procedures for various data structures including lists and arrays [2, 1, 4, 13, 6]. More precisely, a superposition calculus [18] based on rewriting techniques can be used for this purpose. Then, the Nelson-Oppen method can be applied to combine the theory of a data structure with a theory of arithmetic [10]. However, the genuine Nelson-Oppen has a severe limitation since it applies only to signature-disjoint theories. Recently, a non-disjoint extension of the Nelson-Oppen framework has been designed in [8, 9, 15]. In this paper, we show how to use this non-disjoint combination method to build a decision procedure for the union of (1) a (convex) theory modeling some data structure and whose successor function that expresses some counting capabilities, and (2) the linear or non-linear arithmetic over the rationals augmented by the successor function. Both theories share the successor function s and have a common subtheory, called the theory of Increment, axiomatizing the acyclicity and the injectivity of s. This paper is the continuation of a previous work, where we studied the combination of superposition-based decision procedures for the union of two data structures sharing the theory of Integer Offsets [17]. Unfortunately, it was not possible in [17] to integrate standard procedures for reasoning about arithmetic. Here, we focus on the union of theories modeling data structures and fragments of arithmetic sharing the theory of Increment. This union allows us to handle more expressive arithmetic constraints and to obtain a combined decision procedure in which the procedures for the individual theories can be constructed by using an appropriate superposition calculus for data structures but also by classical solving techniques for reasoning about arithmetic (Gauss/Fourier-Motzkin elimination, Groebner bases computation). Our aim is to consider arithmetic constraints over non-necessarily positive numbers, and so the theory of Integer Offsets is not the right axiomatisation. Formally, the theory of Increment is the theory of Integer Offsets minus the axiom $\forall x \;\; 0 \neq \mathsf{s}(x)$, that is true in $\mathbb{N}$ but not in $\mathbb{Z}$ nor in $\mathbb{Q}$. In this paper, we adapt to the theory of Increment the superposition calculus developed for the Integer Offsets in [17]. For the theories we are interested in, we check that all the assumptions for applying the non-disjoint combination method are satisfied. To be effective, the combination method makes use of procedures able to compute the logical

consequences over the shared signature that are exchanged in the main loop of the method. A major contribution of this paper is to build these procedures by using classical solving techniques for arithmetic constraints.

The paper is organized as follows. Section 2 introduces the basic definitions and notations. In Section 3, we present several data structures for which a superposition calculus modulo the theory of Increment can be turned into a decision procedure. In Sections 4 and 5, we present two fragments of arithmetic and the related decision procedures. Section 6 shows how to instantiate the non-disjoint combination method for our need, when the theory of Increment is used as the shared theory. This combination method makes use of procedures for computing all the shared logical consequences to be exchanged. We explain how to compute the required consequences by using the proposed superposition calculus. For the two considered fragments of arithmetic, we show how to compute the needed consequences by using respectively Gauss/Fourier-Motzkin elimination and Groebner bases computation. Eventually, in Section 7, we conclude with some final remarks. Omitted proofs 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 the symbol $\bowtie$ will denote either $=$ or $\neq$. The signature obtained from Σ by adding a set $\underline{a}$ of new constants (i.e., 0-ary function symbols, each of them again equipped with its sort) is denoted by $\Sigma^{\underline{a}}$ and named a *simple expansion* of Σ. 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* $\mathcal{M}$: 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 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 *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 $T \models \varphi(\underline{x}) \leftrightarrow \varphi'(\underline{x})$. A Σ-theory T is *convex* if for any set of Σ-literals and any Σ-atoms $\alpha_1, \ldots, \alpha_n$, we have that $T \models \Gamma \rightarrow (\alpha_1 \vee \cdots \vee \alpha_n)$ implies $T \models \Gamma \rightarrow \alpha_i$ for some i. In the following, all considered theories are convex.

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. Given an idempotent substitution $\sigma = \{x_i \mapsto t_i\}_{i\in I}$, $\widehat{\sigma}$ denotes the constraint $\bigwedge_{i\in I} x_i = t_i$. Note that this constraint in *solved form* is satisfiable in any model. The special symbol $\bot$ denotes a literal which is unsatisfiable in any model.

3 Theories

All the examples of data structures we are interested in involve also a successor function (denoted by s) that satisfies the axioms formalizing the properties of injectivity and acyclicity.

Theory of Increment.

$\boxed{T_S}$ denotes the theory of Increment defining the behaviour of the successor function s and the constant 0. T_S has the mono-sorted signature $\Sigma_S := \{0 : \text{NUM}, \mathsf{s} : \text{NUM} \to \text{NUM}\}$, and it is axiomatized as follows:

$$\begin{array}{ll} \forall x, y & \mathsf{s}(x) = \mathsf{s}(y) \to x = y \\ \forall x & x \neq \mathsf{s}^n(x) \quad \text{for all } n \text{ in } \mathbb{N}^+ \end{array}$$

We consider below some theories T corresponding to standard data structures and we focus on the constraint satisfiability problem for $T \cup T_S$.

Lists.

$\boxed{T_{LS}}$ is a theory of lists endowed with length. The many-sorted signature of T_{LS} is Σ_S plus the set of function symbols $\{\mathsf{nil} : \text{LISTS}, \mathsf{car} : \text{LISTS} \to \text{ELEM}, \mathsf{cdr} : \text{LISTS} \to \text{LISTS}, \mathsf{cons} : \text{ELEM} \times \text{LISTS} \to \text{LISTS}, \ell : \text{LISTS} \to \text{NUM}\}$ and the predicate symbol $\mathsf{atom} : \text{LISTS}$. The axioms[1] of T_{LS} are:

$$\begin{array}{ll} \mathsf{car}(\mathsf{cons}(x, y)) = x & \neg\mathsf{atom}(x) \to \mathsf{cons}(\mathsf{car}(x), \mathsf{cdr}(x)) = x \\ \mathsf{cdr}(\mathsf{cons}(x, y)) = y & \neg\mathsf{atom}(\mathsf{cons}(x, y)) \\ \ell(\mathsf{nil}) = 0 & \mathsf{atom}(\mathsf{nil}) \\ \ell(\mathsf{cons}(x, y)) = \mathsf{s}(\ell(y)) & \end{array}$$

The theory T'_{LS} corresponds to a slight variant of T_{LS} where the sort ELEM coincides with the sort NUM. It is important to notice that, by applying some standard reasoning (see, e.g., [17]), we can substitute T_{LS} (resp. T'_{LS}) with its subset of purely equational axioms, say T_{ELS} (resp. T'_{ELS}), and enrich the set of ground literals G we want to test for satisfiability modulo T_{LS} (resp. T'_{LS}), to a set of literals H in such a way that $T_{LS} \cup G$ is equisatisfiable to $T_{ELS} \cup H$ (resp. $T'_{LS} \cup G$ is equisatisfiable to $T'_{ELS} \cup H$). In this way we can still consider T_{LS} and T'_{LS} as *equational* theories.

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

Records.

$\boxed{T_{RS}}$ denotes a theory of records with increment defined as follows. We consider records in which all the attribute identifiers are associated to the sort NUM or to sorts ELEM_i, and suppose we want to be able to increment by a unity every value of sort NUM stored into the record. To formalize this situation, the signature of T_{RS} is Σ_S plus the function symbols defined as follows. Let $Id = \{id_1, id_2, \ldots, id_n\}$ be a set of attribute identifiers id_i associated to NUM or ELEM_i. Let NI be the set of elements $i \in \{1, \ldots, n\}$ such that id_i is associated to NUM, and let $\overline{NI}$ be $\{1, \ldots, n\} \backslash NI$. Let us name REC the sort of records; for every attribute identifier $id_1, id_2, \ldots, id_n$ we have a couple of functions $\mathsf{rselect_i} : \text{REC} \to \text{NUM}$ and $\mathsf{rstore_i} : \text{REC} \times \text{NUM} \to \text{REC}$ for $i \in NI$; $\mathsf{rselect_i} : \text{REC} \to \text{ELEM}_i$ and $\mathsf{rstore_i} : \text{REC} \times \text{ELEM}_i \to \text{REC}$ for $i \in \overline{NI}$. Moreover, there is also an increment function $\mathsf{incr} : \text{REC} \to \text{REC}$ that increments the elements of sort NUM. The axioms of T_{RS} are:

for every i, j such that $1 \leq i, j \leq n$, $i \neq j$

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

for any $i \in NI$, $\mathsf{rselect_i}(\mathsf{incr}(x)) = \mathsf{s}(\mathsf{rselect_i}(x))$ and for any $i \in \overline{NI}$, $\mathsf{rselect_i}(\mathsf{incr}(x)) = \mathsf{rselect_i}(x)$.

The theory T'_{RS} denotes the particular case where all elements of records are of sort NUM, i.e. $\overline{NI} = \emptyset$. Moreover, following the same argument used in [1], it is possible to check the satisfiability forgetting the extensionality axioms, thus again the theory of records can be still considered as an *equational* one.

Trees.

$\boxed{T_{BS}}$ corresponds to a theory of binary trees endowed with size functions. The many-sorted signature of T_{BS} is Σ_S plus the set of function symbols $\{\mathsf{bin} : \text{ELEM} \times \text{TREES} \times \text{TREES} \to \text{TREES}, \mathsf{null} : \text{TREES}, \mathsf{size}_L : \text{TREES} \to \text{NUM}, \mathsf{size}_R : \text{TREES} \to \text{NUM}\}$. The axioms of T_{BS} are:

$$\begin{aligned} &\mathsf{size}_L(\mathsf{null}) = 0 && \mathsf{size}_R(\mathsf{null}) = 0 \\ &\mathsf{size}_L(\mathsf{bin}(e, t_1, t_2)) = \mathsf{s}(\mathsf{size}_L(t_1)) && \mathsf{size}_R(\mathsf{bin}(e, t_1, t_2)) = \mathsf{s}(\mathsf{size}_R(t_2)) \end{aligned}$$

The function size_L (resp. size_R) computes the length of the left (resp. right) branch of the input binary tree.

The theory T'_{BS} denotes the particular case where ELEM and NUM coincide.

3.1 Decision Procedure using a Superposition Calculus

Consider the theory of Increment T_S defined in Section 3. We want to develop a calculus able to take into account the axioms of T_S in a framework based on superposition. To this aim, we adapt the calculus presented in [17]. 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 three more rules over ground terms presented in Figure 3.

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*, and the following hold: *(i)* $u\sigma \not\preceq t\sigma$, *(ii)* $l[u']\sigma \not\preceq r\sigma$.

Figure 1: Expansion Inference Rules.

We call the so introduced calculus $\mathcal{SP}_S$ and, from now on, we assume that the ordering we consider when performing any application of $\mathcal{SP}_S$ is T_S*-good*.

Definition 1 *We say that an ordering $\succ$ over terms on a signature containing Σ_S is* T_S-good *whenever it satisfies the following requirements:*

(i) $\succ$ is a simplification ordering that is total on ground terms;

(ii) 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).*

It is easy to build a T_S*-good* ordering: for example, it is enough to consider a lexicographic path ordering (LPO) with a precedence $>$ over the symbols in the signature such that $f > \mathsf{s}$ for all the symbols f in the signature different from s.

Theorem 1 *Let T be a Σ-theory presented as a finite set of unit clauses such that $\Sigma \supseteq \Sigma_S$, and assume there is an ordering over terms that is* T_S-good. *$\mathcal{SP}_S$ induces a decision procedure for the constraint satisfiability problem w.r.t. $T \cup T_S$ if, for any set G of ground literals:*

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

Corollary 1 *For any theory $T \in \{T_{LS}, T'_{LS}, T_{RS}, T'_{RS}, T_{BS}, T'_{BS}\}$, $\mathcal{SP}_S$ induces a decision procedure for the constraint satisfiability problem w.r.t. $T \cup T_S$.*

4 Theory of Linear Rational Arithmetic

A very natural extension of the theory of Increment is the linear arithmetic over the rationals. In more detail, let us fix the signature over the sort NUM

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

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

Figure 2: Contraction Inference Rules.

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

where S is a set of literals.

Figure 3: Ground reduction Inference Rules.

$\Sigma_{\mathbb{Q}} := \{0, 1, +, -, \{f_q\}_{q \in \mathbb{Q}}, \mathsf{s}, <\}$, where $0, 1$ are constants, $-$, f_q, s are unary function symbols, $+$ is a binary one and $<$ is a binary predicate symbol. Let $T_{\mathbb{Q}}$ be the set of all the $\Sigma_{\mathbb{Q}}$-sentences that are true in $\mathbb{Q}$ considered as an ordered $\mathbb{Q}$-vector space, under the obvious convention that $0, 1, -, +, <$ are interpreted in their intended meaning, s is the function that to each rational q associates the rational $q + 1$, and the f_q's represent the external product of the $\mathbb{Q}$-vector spaces.

We can observe that in all the models of $T_{\mathbb{Q}}$ the function for the successor function symbol s has an explicit definition using only the symbol 1 and $+$, since $T_{\mathbb{Q}} \models \forall x, y \;\; (y = \mathsf{s}(x) \leftrightarrow y = x + 1)$. This observation can be useful in order to rewrite all the formulae over $\Sigma_{\mathbb{Q}}$ discarding the symbol s.

4.1 Decision Procedure

To build a $T_{\mathbb{Q}}$-satisfiability procedure, a possible solution is to transform equalities and disequalities into inequalities and then to apply the Fourier-Motzkin elimination procedure for checking the satisfiability of the resulting set inequalities. But for efficiency reasons, it is more convenient to keep the initial form of literals. Moreover, we are interested in a decision procedure enhanced with the capability of computing some particular entailed equalities. To this aim, we use the notions of solver and canonizer introduced by Shostak [19]. A solver (*solve*) for $T_{\mathbb{Q}}$ computes a solved form (an idempotent substitution) of a set of equalities given by the Gauss elimination procedure, and a canonizer (*canon*) for $T_{\mathbb{Q}}$ is the classical normalization of arithmetic expressions (assuming an ordering over free constants). Any set of literals denoted by Γ is partitioned into a set of equalities $\Gamma^{=}$, a set of disequalities $\Gamma^{\neq}$ and a set of inequalities $\Gamma^{\leq}$.

Since $T_{\mathbb{Q}}$ is convex, it is possible to take into account disequalities in an easy way. To handle inequalities, we use Fourier-Motzkin elimination to derive (1) unsatisfiable inequalities $q \leq 0$ where q is a strictly positive rational and (2) implicit equalities.

Equalities and Disequalities. The convexity of $T_{\mathbb{Q}}$ justifies the following lemma.

Lemma 1 *Let $\Gamma^{=}$ be a $T_{\mathbb{Q}}$-satisfiable set of equalities, and let $\Gamma^{\neq}$ be a set of disequalities. $\Gamma^{=} \wedge \Gamma^{\neq}$ is $T_{\mathbb{Q}}$-unsatisfiable iff there is some $s \neq t \in \Gamma^{\neq}$ such that $canon(s\gamma) = canon(t\gamma)$, where $\gamma = solve(\Gamma^{=})$.*

Inequalities. It is well-known that Fourier-Motzkin elimination provides a $T_{\mathbb{Q}}$-satisfiability procedure for inequalities. Moreover, it can be slightly adapted to derive "implicit" equalities. Given a set of inequalities $\Gamma^{\leq}$, an inequality $s \leq t$ in $\Gamma^{\leq}$ is an *implicit equality* if $T_{\mathbb{Q}} \models \Gamma^{\leq} \rightarrow s = t$. These implicit equalities can be derived by Fourier-Motzkin elimination and are propagated to the Gauss elimination procedure. The use of Fourier-Motzkin is justified by results expressed in [11, 12] for the case of the reals. These results hold also when the rationals are considered:

- An inequality in $\Gamma^{\leq}$ is an implicit equality iff it appears in a derivation computed by Fourier-Motzkin leading to the inequality $0 \leq 0$.
- If an equality is entailed by $\Gamma^{\leq}$, then it is entailed by the implicit equalities of $\Gamma^{\leq}$.

A $T_{\mathbb{Q}}$-satisfiability procedure can be obtained by using an architecture with the following components:

GE (Gauss) The solver is applied to compute a solved form for the set of equalities. Solved variables are substituted in disequalities and inequalities.

DH (Disequalities Handler) The canonizer is used to check whether a disequality $s \neq t$ is canonized into a trivially unsatisfiable disequality $u \neq u$.

FME (Fourier-Motzkin) Provided that Gauss does not apply, Fourier-Motzkin is used to derived unsatisfiable (ground) inequalities or implicit equalities. Fourier-Motzkin eliminates successively the variables occurring in the inequalities. Eventually, if it derives an inequality $q \leq 0$ such that q is a strictly positive rational, then the unsatisfiability is reported. If it derives an inequality $0 \leq 0$, then the implicit equalities used in the derivation of $0 \leq 0$ are sent to **GE**.

This procedure is terminating because neither **GE** nor **FME** introduces new variables and **GE** strictly decreases the number of unsolved variables. By analysing more precisely this procedure, one can remark that it is sufficient to apply **FME** only once. When we assume that **GE** (resp. **FME**) is applied on the whole set of equalities (resp. inequalities), sending to **GE** the implicit equalities found by **FME** does not help to find further implicit equalities by applying **FME** again, but it computes a solved form used to check the satisfiability of disequalities.

It is easy to show the correctness and completeness of this procedure. Indeed, if $false$ is derived then the input is unsatisfiable (since all inferences are obviously correct). Otherwise, it produces eventually a conjunction of the form $\widehat{\sigma} \wedge \Phi^{\neq} \wedge \Phi^{\leq}$ such that (1) $\widehat{\sigma}$ is a solved form such that every variable in the domain of the substitution σ occurs only once in $\widehat{\sigma} \wedge \Phi^{\neq} \wedge \Phi^{\leq}$, (2) $\Phi^{\neq}$ is a set of disequalities such that for any $s \neq t \in \Phi^{\neq}$, $canon(s) \neq canon(t)$, and (3) $\Phi^{\leq}$ is a $T_{\mathbb{Q}}$-satisfiable set of inequalities containing no implicit equalities. Thanks to (3), $\Phi^{\neq} \wedge \Phi^{\leq}$ is $T_{\mathbb{Q}}$-satisfiable too, and then by (1), we can conclude that $\widehat{\sigma} \wedge \Phi^{\neq} \wedge \Phi^{\leq}$ is $T_{\mathbb{Q}}$-satisfiable.

5 Theory of $\mathbb{Q}$-Algebras

We can consider now another extension of the theory of Increment, namely we can see T_S as subtheory of the theory of (non-degenerate) $\mathbb{Q}$-algebras. More in detail, we fix as a signature $\Sigma_{\mathbb{Q}\text{-}alg}$ the set consisting in the constants $0, 1$, the two binary function symbols $+, \times$, the unary function symbols $-$ and the $\mathbb{Q}$-indexed family of unary function symbols f_q. As a notational convention, of course we use the infix notation for $+$ and write qv, $v_1 v_2$ for $f_q(v)$, $\times(v_1, v_2)$, respectively. The theory of $\mathbb{Q}$-algebras, denoted by $T_{\mathbb{Q}\text{-}alg}$, is described using the axioms of abelian groups for $+$ (stating the associativity, the commutativity of $+$, the existence of the inverse $-v$ for each v and the fact that 0 is the unity of $+$), the axioms of abelian monoids for $\times$ (asserting the associativity and the commutativity of $\times$, and that 1 is the unity of $\times$), the fact that 0 is different from 1 and the other six axioms relating the behaviour of $+$ and $\times$

for every q, q_1 and q_2 in $\mathbb{Q}$

$$\forall x, y, z\ (x + y)z = xz + yz \tag{1}$$
$$\forall x, y\ q(x + y) = qx + qy \tag{2}$$
$$\forall x\ (q_1 \oplus q_2)x = q_1 x + q_2 x \tag{3}$$
$$\forall x\ (q_1 \cdot q_2)x = q_1(q_2 x) \tag{4}$$
$$\forall x\ 1_{\mathbb{Q}} x = x \tag{5}$$
$$\forall x, y\ q(xy) = x(qy) \tag{6}$$

where $\oplus$ and $\cdot$ are respectively the sum and multiplication operation in $\mathbb{Q}$, and $1_{\mathbb{Q}}$ is the multiplicative unit of $\mathbb{Q}$.

Again, the symbol s admits in $T_{\mathbb{Q}\text{-}alg}$ the explicit definition as in the previous example: we have $T_{\mathbb{Q}\text{-}alg} \models \forall x, y\ (y = \mathsf{s}(x) \leftrightarrow y = x + 1)$. Injectivity of s is guaranteed by the group structure (i.e., it holds $T_{\mathbb{Q}\text{-}alg} \models \forall x, y\ (x+1 = y+1 \leftrightarrow x = y)$), and the acyclicity of s is guaranteed by the fact that $1 \neq 0$ and by the axiom (4).

5.1 Decision Procedure

Given a set $\underline{a}$ of n fresh constants, the ground atoms over $\Sigma^{\underline{a}}_{\mathbb{Q}\text{-}alg}$ are polynomials in at most n indeterminates whose normalized representation is of the kind $p(\underline{a}) = 0$. Given the convexity of $T_{\mathbb{Q}\text{-}alg}$, The constraint satisfiability problem in $T_{\mathbb{Q}\text{-}alg}$ is just the problem of deciding whether an equation $p(\underline{a}) = 0$ is a

logical consequence of a finite number of equations $\{p_1(\underline{a}) = 0, \ldots, p_m(\underline{a}) = 0\}$. Since the polynomial ring $\mathbb{Q}[a_1, \ldots, a_n]$ is the free $\mathbb{Q}$-algebra over n generators, this problem is equivalent to the membership of the polynomial p to the ideal $\langle p_1, \ldots, p_m \rangle$ generated by the polynomials $p_1, \ldots, p_m$. The Buchberger algorithm solves the problem by computing the Groebner basis associated to the ideal $\langle p_1, \ldots, p_m \rangle$ [5].

6 Non-Disjoint Combination of Theories

It would be interesting for us to take into account constraints that involve symbols used to describe the data structures and symbols for the arithmetic. Usually, these constraints are handled relying on a framework that allows to combine the already available decision procedures for the theories of the data structures and the arithmetic, provided that the theories are formalized on signatures that share only the equality predicate symbol.

In the examples we have considered in Section 3, if we imagine to deal with constraints involving also arithmetical symbols, the requirement for the signatures to be disjoint cannot be satisfied, since every theory in the above examples presents some axioms involving the successor function symbol s. Note that, even if for the arithmetic the symbol s may not be used, in order to have a meaningful answer for the satisfiability of constraints, it is necessary to recall the axiom that links s and $+$, i.e. the axioms that defines s by the formula $\forall x, y\, (y = \mathsf{s}(x) \leftrightarrow y = x + 1)$.

At this point, we have at our disposal satisfiability procedures for $T_{\mathbb{Q}}$, $T_{\mathbb{Q}\text{-}alg}$ and for the theories of some data structures. Since all these theories share the theory T_S over the signature $\Sigma_S = \{0, \mathsf{s}\}$, we look if the general framework for the combination of non-disjoint theories developed in [9] could be applied. This framework extends the well-known Nelson-Oppen methodology, and can guarantee, under the conditions described in the following, the transfer of the decidability of the satisfiability problem.

Theorem 2 *[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.

We will not enter into the detail of the conditions required by the theorem; we will simply specialize them when the theory of Increment T_S plays the role of the shared theory between a theory modelling a data structure as in Section 3 and the arithmetic over the rationals (in both the cases $T_{\mathbb{Q}}$ and $T_{\mathbb{Q}\text{-}alg}$).

T_S is a universal theory; moreover, if we add to T_S the axiom $\forall x \exists y \;\; x = \mathsf{s}(y)$, we obtain a theory $T_S^\star$ that admits quantifier elimination (it is easy, e.g., to adapt the procedure in [7]) and such that every constraint that is satisfiable in a model of T_S is satisfiable also in a model of $T_S^\star$. To justify the last claim, it is sufficient to observe that each model of T_S can be extended to a model of $T_S^\star$ simply by adding recursively to each element a "predecessor". Now, for any theory $T \supseteq T_S$ over a signature $\Sigma \supseteq \Sigma_S$ the T_S-compatibility requirement simply reduces to the following definition.

Definition 2 (T_S-compatibility) *Let T be a theory in the signature $\Sigma \supseteq \Sigma_S$. We say that T is* T_S-compatible *iff $T_S \subseteq T$ and every Σ-constraint which is satisfiable in a model of T is satisfiable also in a model of $T_S^\star \cup T$.*

In our case, Definition 2 requires that the satisfiability problem has the same answer in the models of T and in the models of $T \cup \{\forall x \exists y \;\; x = \mathsf{s}(y)\}$.

It is immediate now to verify that $T_{\mathbb{Q}}$ is a T_S-compatible theory, since all the models of $T_{\mathbb{Q}}$ are already models of $T_S^\star$: indeed, for each element r in a model of $T_{\mathbb{Q}}$, the (unique) element t such that r is equal to the (interpretation of the) successor of t is simply (the interpretation of) $r-1$. The same kind of argument can be applied also to show the T_S-compatibility of $T_{\mathbb{Q}\text{-}alg}$, since, in each model of $T_{\mathbb{Q}\text{-}alg}$, for each element t the (unique) element v such that t is equal to the (interpretation of the) successor of v is again (the interpretation of) $t-1$. As far as the other theories presented in Section 3 for the lists, the records and the trees are concerned, it is easy to see that the T_S-compatibility requirement holds again, because the eventual adjunction of predecessors to the elements in the sort NUM does not affect the satisfiability of constraints; more details can be found in [17].

Let us analyze the third requirement of Theorem 2. Roughly speaking, the property of being Noetherian for T_S means that, fixed a finite number of fresh constants, there exists only a finite number of Σ_S-atoms over those constants that are not redundant when reasoning modulo T_S.

Definition 3 (Noetherian Theory) *T_S 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 $\Sigma_S^{\underline{a}}$-atoms is eventually constant modulo T_S, i.e. there is an n such that $T_S \cup \Theta_n \models \alpha$, for every natural number m and atom $\alpha \in \Theta_m$.*

Since it is possible to prove (see, e.g. [20]) that all the theories whose signature contains only constants and one unary function symbol are Noetherian, it follows that the theory of Increment T_S enjoys this property.

Exactly as it happens for the original Nelson-Oppen procedure, the result of Theorem 2 strongly relies on the capability of deducing logical consequences over the shared signature. To this aim, let us consider a convex theory $T \supseteq T_S$ with signatures $\Sigma \supseteq \Sigma_S$, and suppose we want to discover, given an arbitrary set of ground literals Γ over Σ, a "complete set" of logical positive consequences of Γ over Σ_S, formalized by the notion of *T_S-basis*.

Definition 4 (T_S-basis) *Given a convex Σ-theory $T \supseteq T_S$ and a finite set Γ of ground literals (built out of symbols from Σ and possibly further free constants) and a finite set of free constants $\underline{a}$, a* T_S-basis *modulo T for Γ w.r.t. $\underline{a}$ is a set Δ of ground $\Sigma_S^{\underline{a}}$-atoms such that*

(i) $T \cup \Gamma \models \alpha$, *for all* $\alpha \in \Delta$ *and*

(ii) if $T \cup \Gamma \models \alpha$ *then* $T_S \cup \Delta \models \alpha$, *for every ground* $\Sigma_S^{\underline{a}}$*-atom* α.

The Noetherianity of T_S guarantees that, for every set of Σ-literals Γ and for every set $\underline{a}$ of constants, a finite T_S-basis Δ for Γ w.r.t. $\underline{a}$ always exists. Note that if Γ is T-unsatisfiable then w.l.o.g. $\Delta = \{\bot\}$. Unfortunately, a basis does not need to be computable; this motivates the following definition corresponding to the last hypothesis of Theorem 2.

Definition 5 *A convex theory* T *is an* effectively Noetherian extension of T_S *if and only if* T_S *is Noetherian and a* T_S*-basis modulo* T *is computable for every set of literals and 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 2, and that extends the Nelson-Oppen combination method to theories over non disjoint signatures. In the algorithm below, 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_S-$\mathsf{basis}_{T_i}(\Gamma_i) = \Delta_i$ and $\bot \notin \Delta_i$ 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. Return *"satisfiable"*.

The requirement of being effectively Noetherian extension of T_S for $T_{\mathbb{Q}}$, $T_{\mathbb{Q}\text{-}alg}$ and the theories of the data structures in Section 3 is the last condition that remains to be guaranteed. In the following we show how the decision procedures that we have already presented can be used to this aim.

6.1 Computing T_S-bases for Data Structures

In this section we show that the superposition calculus $\mathcal{SP}_S$ allows us to build T_S-bases modulo theories that are axiomatized by unit clauses.

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 $\underline{a}, \underline{b}$. The theory $T \cup T_S$ is convex because it is a Horn theory. At this point, Proposition 1 shows how $\mathcal{SP}_S$ can be used in order to derive T_S-bases.

Proposition 1 *Let* S_ω *be a finite saturation of* $T \cup G(\underline{a}, \underline{b})$ *w.r.t* $\mathcal{SP}_S$ *using a* T_S-good *order over the terms in the signature* $\Sigma \cup \{\underline{a}, \underline{b}\}$ *such that* (i) *every term over the subsignature* $\Sigma_S^{\underline{a}}$ *is smaller than any term that contains a symbol in* $(\Sigma \setminus \Sigma_S) \cup \{\underline{b}\}$, (ii) *not containing* $\bot$, *and such that* (iii) s*-rooted terms can be maximal just in ground equations in* S_ω *and* (iv) *variables of sort* NUM *are never the maximal term in the equations. The set* $\Delta(\underline{a})$ *of all the ground equations over* $\Sigma_S^{\underline{a}}$ *in* S_ω *is a* T_S*-basis for* T.

Corollary 2 $\mathcal{SP}_S$ *is able to compute* T_S*-bases for the theories* $T \cup T_S$, *where* T *varies in* $\{T_{LS}, T'_{LS}, T_{RS}, T'_{RS}, T_{BS}, T'_{BS}\}$ *as presented in Section 3.*

6.2 Computing T_S-bases for Fragments of Arithmetic

In this section we will show how to derive T_S-bases when we consider the theory $T_{\mathbb{Q}}$ and the theory $T_{\mathbb{Q}\text{-}alg}$. First of all, we recall that both $T_{\mathbb{Q}}$ and $T_{\mathbb{Q}\text{-}alg}$ are convex theories and we will see that, in both the cases, given a set of atoms, the respective decision procedures are able to derive a 'representative set' of the linear equalities, i.e. equalities in the shape $q_1x_1 + \cdots + q_nx_n = 0$, $q_i \in \mathbb{Q}$, that are implied.

Our aim is, at that point, to describe a procedure that, given a generic constraint over $\Sigma_{\mathbb{Q}}$ (resp. $\Sigma_{\mathbb{Q}\text{-}alg}$), say Γ, is able to derive a set of ground atoms over an expansion Σ_S^a, say Δ, such that $T_{\mathbb{Q}} \cup \Gamma \models \Delta$ (resp. $T_{\mathbb{Q}\text{-}alg} \cup \Gamma \models \Delta$), and such that, for every Σ_S^a-atom e it holds that $T_{\mathbb{Q}} \cup \Gamma \models e$ iff $T_S \cup \Delta \models e$ (resp. $T_{\mathbb{Q}\text{-}alg} \cup \Gamma \models e$ iff $T_S \cup \Delta \models e$).

We start by recalling that all the literals in Γ that are not atoms, i.e. that are the negation of some atoms, are irrelevant in order to compute the set Δ.

Lemma 2 *Let T be a convex theory, let P be a set of atoms, let N be a set of negative literals, i.e. a set consisting only of negations of atoms, and let α be an atom. If $P \wedge N$ is T-satisfiable, it holds $T \models (P \wedge N) \rightarrow \alpha$ iff $T \models P \rightarrow \alpha$.*

Let us now introduce $T_{\mathbb{Q}}^{=}$, the theory of the (non-degenerate) $\mathbb{Q}$-vector spaces. This theory is a subtheory of both $T_{\mathbb{Q}}$ and $T_{\mathbb{Q}\text{-}alg}$, it is built on the signature $\Sigma_{\mathbb{Q}^=} := \{0, 1, +, -, \{f_q\}_{q\in\mathbb{Q}}, \mathsf{s}\}$, and it is ruled by the axioms of abelian groups over the $+$, the requirement that $1 \neq 0$ and the axioms (3) – (6) in Section 5. Again, we require the relationship $\forall x, y\ (y = \mathsf{s}(x) \leftrightarrow y = x + 1)$ to hold in all the structure that are models of $T_{\mathbb{Q}}^{=}$.

Lemma 3 *Let $\underline{a}$, $\underline{b}$ be two sets of free constants such that $\underline{a} \subseteq \underline{b}$. Given a $T_{\mathbb{Q}}^{=}$-satisfiable set of linear equalities P over the signature $\Sigma_{\mathbb{Q}^=}^{\underline{b}}$, it is possible to derive a T_S-basis modulo $T_{\mathbb{Q}}^{=}$ for P w.r.t. $\underline{a}$.*

Proof. Any Σ_S^a-equation is of the form $\mathsf{s}^{n_1}(a_1) = \mathsf{s}^{n_2}(a_2)$ for some n_1, n_2 in $\mathbb{N}$ and for some a_1, a_2 in $\underline{a} \cup \{0\}$. Due to the injectivity axiom for the s function symbol, any equation can be equivalently rewritten in the form $a_1 = \mathsf{s}^{n_2-n_1}(a_2)$ whenever $n_2 \geq n_1$, or in the form $\mathsf{s}^{n_1-n_2}(a_1) = a_2$ whenever $n_1 \geq n_2$. Thus, for any couple of constants a_1, a_2 in $\underline{a}$, it is sufficient to detect if $T_{\mathbb{Q}}^{=} \cup P \models a_1 = a_2 + n$ for some $n \in \mathbb{N}$, or if $T_{\mathbb{Q}}^{=} \cup P \models a_2 = a_1 + n$ (for some $n \in \mathbb{N}$, again). While running the Gauss elimination procedure on P and computing $\sigma = solve(P)$, we obtain:

$$T_{\mathbb{Q}}^{=} \cup P \models a_1 = a_2 + n \text{ iff } canon(a_1\sigma - a_2\sigma) = n$$

Let Δ be the set of Σ_S^a-equations obtained by collecting all the equations of the form $a_1 = \mathsf{s}^n(a_2)$ for which $canon(a_1\sigma - a_2\sigma) = n$. The properties (i) and (ii) of Definition 4 for T_S-bases are straightforward.

Example 1 *Consider $P = \{a_1 - 1 = a_3 + 1, 2b_2 + a_3 = b_2 + 2b_1 + b_2, a_2 - 1 = 2a_3 - 2b_1\}$. A solved form for P is given by $\sigma = \{a_1 \mapsto 2b_1 + 2, a_2 \mapsto 2b_1 + 1, a_3 \mapsto 2b_1\}$. By using the method given in the proof of Lemma 3, we can derive that $a_1 = \mathsf{s}^2(a_3), a_2 = \mathsf{s}(a_3)$ and these equalities define a T_S-basis modulo $T_{\mathbb{Q}}^{=}$ for P w.r.t. $\{a_1, a_2, a_3\}$.*

6.2.1 The $T_{\mathbb{Q}}$ case.

While running over a constraint Γ the procedure presented in Section 4.1, we have already pointed out that, if Γ is satisfiable, the procedure halts returning a conjunction of the form $\widehat{\sigma} \wedge \Phi^{\neq} \wedge \Phi^{\leq}$, where $\widehat{\sigma}$ is a set of linear equalities that, thanks to the results in [11, 12] and Lemma 2, satisfies the following two properties:

1. $T_{\mathbb{Q}} \cup \Gamma \models \widehat{\sigma}$;
2. if e is a linear equality such that $T_{\mathbb{Q}} \cup \Gamma \models e$, then $T_{\mathbb{Q}}^{=} \cup \widehat{\sigma} \models e$.

6.2.2 The $T_{\mathbb{Q}\text{-}alg}$ case.

In Section 5.1, we have recalled that the satisfiability problem modulo $T_{\mathbb{Q}\text{-}alg}$ can be solved by running the Buchberger algorithm for computing the Groebner basis associated to a set Γ of polynomials. Actually, the Groebner basis computation can be considered as a way to obtain a confluent and terminating rewriting system for deciding the universal fragment of the theory of $\mathbb{Q}$-algebras. In [15], it is shown how a little tuning on the ordering of the rules in the term rewriting system is able to produce in the final Groebner basis associated to Γ a set, say P, of linear polynomials such that:

1. $T_{\mathbb{Q}\text{-}alg} \cup \Gamma \models P$;
2. if e is a linear polynomial such that $T_{\mathbb{Q}\text{-}alg} \cup \Gamma \models e$, then $T_{\mathbb{Q}}^{=} \cup P \models e$.

Proposition 2 *Let $\underline{a}$, $\underline{b}$ be two sets of free constants such that $\underline{a} \subseteq \underline{b}$. Given a constraint Γ over the signature $\Sigma_{\mathbb{Q}}^{\underline{b}}$ (resp. $\Sigma_{\mathbb{Q}\text{-}alg}^{\underline{b}}$, $(\Sigma_{\mathbb{Q}} \cup \Sigma_{\mathbb{Q}\text{-}alg})^{\underline{b}}$), it is possible to compute a T_S-basis modulo $T_{\mathbb{Q}}$ (resp. $T_{\mathbb{Q}\text{-}alg}$, $T_{\mathbb{Q}} \cup T_{\mathbb{Q}\text{-}alg}$) for Γ w.r.t. $\underline{a}$.*

Proof.

$\boxed{T_{\mathbb{Q}}}$ Let us run the decision procedure for testing the satisfiability of Γ w.r.t. $T_{\mathbb{Q}}$. If it reports unsatisfiability, then the T_S-basis is simply $\bot$. Otherwise collect all the implicit equalities (say $\widehat{\sigma}$) as described in Section 4.1, and apply on $\widehat{\sigma}$ the procedure described in Lemma 3. Thanks to the properties 1. and 2. recalled in the paragraph above about the $T_{\mathbb{Q}}$ case, the set Δ is a T_S-basis. Indeed, since $T_{\mathbb{Q}} \cup \Gamma \models \widehat{\sigma}$ and $T_{\mathbb{Q}}^{=} \cup \widehat{\sigma} \models \Delta$, it follows (i) $T_{\mathbb{Q}} \cup \Gamma \models \Delta$ (recall that $T_{\mathbb{Q}}^{=} \subset T_{\mathbb{Q}}$); moreover it holds the following chain of implications: for any e s.t. $T_{\mathbb{Q}} \cup \Gamma \models e$, then the set of equalities $\widehat{\sigma}$ derived using Fourier-Motzkin and Gauss elimination procedures is such that $T_{\mathbb{Q}}^{=} \cup \widehat{\sigma} \models e$, and thus, by Lemma 3, also (ii) $T_S \cup \Delta \models e$.

$\boxed{T_{\mathbb{Q}\text{-}alg}}$ The case to compute a T_S-basis for Γ is analogous, taking into account the fact that the set P of representative linear polynomials is given by running the Buchberger algorithm as described in [15], and again the properties 1. and 2. in the paragraph above about the $T_{\mathbb{Q}\text{-}alg}$ case.

$\boxed{T_{\mathbb{Q}} \cup T_{\mathbb{Q}\text{-}alg}}$ The proofs of Lemma 3 and the two cases above make clear that, once we are able to guarantee the derivation of a set of linear equations P that satisfy the properties of the kind 1. and 2., we are also able to

compute T_S-bases. Since it is possible to isolate such a set w.r.t. $T_{\mathbb{Q}}$ and $T_{\mathbb{Q}\text{-}alg}$, it is possible to apply Theorem 1.3.12 in [20] to derive, given a set of literals Γ over $(\Sigma_{\mathbb{Q}} \cup \Sigma_{\mathbb{Q}\text{-}alg})^{\underline{b}}$, a set P' of linear equalities such that, again,

1. $T_{\mathbb{Q}} \cup T_{\mathbb{Q}\text{-}alg} \cup \Gamma \models P'$;
2. if e is a linear equality such that $T_{\mathbb{Q}} \cup T_{\mathbb{Q}\text{-}alg} \cup \Gamma \models e$, then $T_Q^{=} \cup P' \models e$.

At this point, it is immediate to apply again Lemma 3 to compute a T_S-basis modulo $T_{\mathbb{Q}} \cup T_{\mathbb{Q}\text{-}alg}$.

6.3 Applying the Combination Method

At the beginning of Section 6, we have pointed out that the theory of Increment T_S is Noetherian and that it can be "enlarged" to T_S^*, which admits quantifier elimination and behaves the same w.r.t. the satisfiability of constraints; moreover we have also shown that $T_{\mathbb{Q}}, T_{\mathbb{Q}\text{-}alg}$ and all the theories for the data structures we have introduced in Section 3 are T_S-compatible. Since the T_S-compatibility is a modular property (cf. Proposition 4.4 in [8]), also $T_{\mathbb{Q}} \cup T_{\mathbb{Q}\text{-}alg}$[2] is T_S-compatible. Moreover, in Section 6.1 we have shown how to compute T_S-bases modulo the theories for the considered data structures, and in Section 6.2 we have shown how to compute T_S-bases modulo the three fragments of arithmetic we are taking into account. Hence, all the hypotheses of Theorem 2 are satisfied.

Theorem 3 *Let DST be the set of theories $\{T_{LS}, T'_{LS}, T_{RS}, T'_{RS}, T_{BS}, T'_{BS}\}$ defined in Section 3. For any Σ_1-theory $T_1 \in DST$ and any Σ_2-theory $T_2 \in \{T_{\mathbb{Q}}, T_{\mathbb{Q}\text{-}alg}, T_{\mathbb{Q}} \cup T_{\mathbb{Q}\text{-}alg}\} \cup DST$ such that $\Sigma_1 \cap \Sigma_2 = \Sigma_S$, $T_1 \cup T_S \cup T_2$ has a decidable constraint satisfiability problem.*

7 Conclusion

We have presented a way to instantiate the non-disjoint extension of the Nelson-Oppen method in order to combine various theories corresponding to data structures with some fragments of arithmetic. Our approach allows us to consider arbitrary arithmetic constraints even if the shared signature is restricted to the successor function. We have focused on fragments over the rationals, but the same results hold in the case we replace the rationals with the reals. On the other hand, the fragments over the integers are more problematic, since first of all the convexity is lost, and secondly it is not so clear how to extract from the existing decision procedures the sets that are representative of the logical consequences involving only the successor function symbol. This is a problem left for future work.

Another interesting issue is to study how to handle more complex connecting axioms between the data structure and the arithmetic, and to try to enlarge the shared signature. In [17], the shared theory is a more precise approximation of the theory of integers, but on the other hand there is no integration of standard techniques for reasoning about arithmetic. In [16], we show how to combine

[2] The satisfiability problem w.r.t. $T_{\mathbb{Q}} \cup T_{\mathbb{Q}\text{-}alg}$ can be decided through an appropriate application of Theorem 2: for the details we refer to [15].

data structures sharing the theory of abelian groups. In a similar way to what is investigated here, it would be interesting to study the combination of a data structure with some fragments of arithmetic when the shared theory is the one of abelian groups.

References

[1] A. Armando, M.-P. Bonacina, S. Ranise, and S. Schulz. New results on rewrite-based satisfiability procedures. *ACM Trans. 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] L. Bachmair and H. Ganzinger. Rewrite-based equational theorem proving with selection and simplification. *Journal of Logic and Computation*, 4(3):217–247, 1994.

[4] M. P. Bonacina and M. Echenim. On variable-inactivity and polynomial T-satisfiability procedures. *Journal of Logic and Computation*, 18(1):77–96, 2008.

[5] B. Buchberger. A theoretical basis for the reduction of polynomials to canonical forms. *ACM SIGSAM Bull.*, 10(3):19–29, 1976.

[6] 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.

[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 D. V. Hung and M. Wirsing, editors, *Proc. of ICTAC 2005*, volume 3722 of *LNCS*, pages 594–608, Hanoi (Vietnam), 2005. Springer-Verlag.

[11] J.-L. Lassez and M. J. Maher. On Fourier's algorithm for linear arithmetic constraints. *Journal of Automated Reasoning*, 9(3):373–379, 1992.

[12] J.-L. Lassez and K. McAloon. A canonical form for generalized linear constraints. *Journal of Symbolic Computation*, 13(1):1–24, 1992.

[13] C. Lynch and D.-K. Tran. Automatic decidability and combinability revisited. In F. Pfenning, editor, *Proc. of CADE'07*, volume 4603 of *LNCS*, pages 328–344, Bremen, Germany, 2007.

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

[15] E. Nicolini. *Combined decision procedures for constraint satisfiability*. PhD thesis, Università degli Studi di Milano, 2007.

[16] E. Nicolini, C. Ringeissen, and M. Rusinowitch. Combinable extensions of abelian groups. In R. Schmidt, editor, *Proc. of CADE'09*, volume 5663 of *LNAI*, pages 51–66, Montreal (Canada), 2009. Springer.

[17] 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. Also as INRIA Report RR-6697.

[18] 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.

[19] R. E. Shostak. Deciding combinations of theories. *J. of the ACM*, 31:1–12, 1984.

[20] D. Zucchelli. *Combination methods for software verification*. PhD thesis, Università degli Studi di Milano and Université Henri Poincaré - Nancy 1, 2008.

A Superposition modulo The Theory of Increment

In this section, for the sake of completeness, we give the proof of the correctness of $\mathcal{SP}_S$, which is a straightforward adaptation of the proof that can be found in [17] for the calculus developed to handle the theory of Integer Offsets. The main difference between the calculus proposed here and the one presented in [17] relies on the rule introduced in [17] to handle the axiom $\forall x\ (0 \neq \mathsf{s}(x))$; here that rule is inappropriate, since the models for the theory of Increment may contain an element that is the predecessor of (the interpretation of) 0. We will see that, if the rule is suppressed, the calculus remains still refutationally complete w.r.t. the models of the theory of Increment under suitable conditions.

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

Definition 6 *Let $\mathcal{SP}_S$ be the calculus depicted in Figures 1, 2 and 3. A derivation (δ) with respect to $\mathcal{SP}_S$ 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 by 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 by removing a literal according to one of the rules in Figure 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 $\{E \mid E \in GS\ \&\ E < L\sigma\} \models L\sigma$. 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.

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 $\{E \mid E \in GS\ \&\ E < C_m\sigma\} \models D\sigma$, 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. $\mathcal{SP}_S$ 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_S. Thus:

Proposition 3 *If the set of persistent literals S_ω contains $\bot$, then S_ω is unsatisfiable in any model of T_S.*

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 4 *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_S (namely the structures in which the function `s` is injective and acyclic.

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_S 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_S) if and only if its saturation S_ω does not contain $\bot$.*

Proposition 5 *Assuming T_S-*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 equations whose maximal term is a variable of sort* NUM*,*
- `s`*-rooted terms can be maximal just in ground equations in S_ω*

then S_ω is satisfiable in a model of T_S.

Proof.

By Proposition 4 we know that if $\bot$ is not derived, then it is possible to build a model $\mathcal{M}$ that satisfies all the literals contained in the limit of the derivation, S_ω. We can build such a model $\mathcal{M}$ adapting to our case the so called *model-generation* technique [3]. By assumption, S_ω contains only literals, so $\mathcal{M}$ will be built over the Herbrand universe relying upon a convergent rewriting system $\mathcal{R}$ defined as follows: suppose that $\mathcal{R}_{\leq D}$ has already been defined for every ground literal D in GS_ω such that $D < C$, and let $\mathcal{R}_{<C} := \bigcup\{\mathcal{R}_{\leq D} \mid D \in GS_\omega \,\&\, D < C\}$. $\mathcal{R}_{\leq C}$ is equal to $\mathcal{R}_{<C} \cup \{l \to r\}$ if

- C is $l = r$;
- l is in normal form with respect to $\mathcal{R}_{<C}$;
- $l > r$.

If any of the above condition is not satisfied, then $\mathcal{R}_{\leq C} := \mathcal{R}_{<C}$.

Thus, given two ground terms t_1 and t_2, $\mathcal{M} \models t_1 = t_2$ if and only if $t_1 \downarrow_{\mathcal{R}} = t_2 \downarrow_{\mathcal{R}}$.

What remains to be shown is that the model so obtained is a structure that satisfies also the axioms of T_S.

In the following, we will call OGS_ω the set of all ground literals that are contained in S_ω. Notice that in OGS_ω both the left and the right side of the literals are inter-reduced. Indeed, by contradiction, suppose that $t = s$ is in OGS_ω and that there exists a rule $l \to r$ in $\mathcal{R}$ that is able to reduce (say) t. $l \to r$ is a ground instance of some equation in S_ω, that means that the rule Simplification should have been applied, deleting thus $t = r$ in S_ω.

We have to prove now that in $\mathcal{M}$ the axioms for the injectivity of (the interpretation of) s and its acyclicity are true.

1) $\forall x, y\, \mathsf{s}(x) = \mathsf{s}(y) \to x = y$

   By contradiction, let us suppose that there exist two terms t_1 and t_2 such that $\mathsf{s}(t_1) \downarrow_{\mathcal{R}} = \mathsf{s}(t_2) \downarrow_{\mathcal{R}}$ but such that $t_1 \downarrow_{\mathcal{R}} \neq t_2 \downarrow_{\mathcal{R}}$. Without loss of generality, we can choose such a pair minimal with respect to the componentwise order over pairs induced by the ordering over the terms. By minimality and by the fact that $\mathcal{R}$ is convergent, we can suppose that both t_1 and t_2 are irreducible. This latter assumption implies that there exist rules in $\mathcal{R}$ such that $\mathsf{s}(t_1) \to r \to^* z$ and $\mathsf{s}(t_2) \to^* z$. Since the rule $\mathsf{s}(t_1) \to r$ belongs to $\mathcal{R}$, the literal $\mathsf{s}(t_1) = r$ belongs to GS_ω. More precisely, it belongs to OGS_ω, since in S_ω there is no non-ground literal that allows to rewrite terms whose root symbol is s. Now two cases are possible:

   - either $\mathsf{s}(t_2)$ is irreducible by $\mathcal{R}$. Then $\mathsf{s}(t_2) \equiv z$, and, by the fact that r is irreducible, we obtain that $r \equiv \mathsf{s}(t_2)$. Therefore, OGS_ω contains the equation $\mathsf{s}(t_1) = \mathsf{s}(t_2)$, that is impossible since an application of the rule R1 would have deleted it and replaced with $t_1 = t_2$;
   - or there is a term r' and a rule $\mathsf{s}(t_2) \to r'$ such that $\mathsf{s}(t_2) \to r' \to^* z$. Again, the equation $\mathsf{s}(t_2) = r'$ belongs to OGS_ω, implying that r' is irreducible. As a consequence $r \equiv r'$. Again, we have a contradiction because an application of the rule R2 would have been possible, deleting (say) $\mathsf{s}(t_1) = r$ and substituting it with $t_1 = t_2$.

2) $\mathsf{s}^n(t) \neq t$ for all the terms t and for all the natural $n \in \mathbb{N}^+$

   By contradiction, there exists a ground term t and a natural m such that $\mathsf{s}^m(t) \downarrow_{\mathcal{R}} t$. We can choose t as the least ground term with that property; by minimality, we have that t is irreducible. Thus it happens that $\mathsf{s}^m(t) \to r_1 \to^* t$ where $\mathsf{s}^m(t)$ reduces to a term r_1 thanks to an application of a rule of the kind $\mathsf{s}^{m_1}(t) \to r$ that comes from the equation $\mathsf{s}^{m_1}(t) = r$ in OGS_ω because only the equations that are in OGS_ω can reduce terms whose root symbol is s. Since t is irreducible, we must have $m_1 > 0$; moreover r is not s-rooted since, otherwise, R1 would be applied, deleting thus $\mathsf{s}^{m_1}(t) = r$. Since r is not s-rooted and by the requirement over $\succ$, $\mathsf{s}^{m_1}(t) \succ r$ implies that $t \succ r$. More in detail, w.l.o.g. we can suppose that $t \equiv \mathsf{s}^n(t')$, where t' is not s-rooted. Due to the requirement over $\succ$ and the fact that r is not s-rooted, we have for every k in $\mathbb{N}$, $\mathsf{s}^k(t') \succ r$ iff $t' \succ r$. In particular, $t \equiv \mathsf{s}^n(t') \succ r$ implies that $t' \succ r$. Now we know that $\mathsf{s}^m(t) \to \mathsf{s}^{m-m_1}(r) \to^* t$; but then

$\mathsf{s}^{m-m_1}(r) \succeq t \equiv \mathsf{s}^n(t')$. Again, $\mathsf{s}^{m-m_1}(r) \succeq \mathsf{s}^n(t')$ iff either $r \succ t'$, that cannot be since $t' \succ r$, or $r \equiv t'$ and $m - m_1 \geq n$. But, if $r \equiv t'$, the equation $\mathsf{s}^{m_1}(t) = r$ in OGS_ω becomes $\mathsf{s}^{m_1+n}(t') = t'$, and, at this point, an application of the rule C1 would have added $\bot$.

Moreover, assume that $G(\underline{a}, \underline{b})$ is a set of ground literals over an expansion of $\Sigma \supseteq \Sigma_S$ with the finite sets of fresh constants $\underline{a}, \underline{b}$. We recall that, being $T \cup T_S$ a Horn theory, the theory $T \cup T_S$ is convex. At this point, Proposition 1 shows how $\mathcal{SP}_S$ can be used in order to derive T_S-bases:

Proposition 1. *Let S_ω be a finite saturation of $T \cup G(\underline{a}, \underline{b})$ w.r.t $\mathcal{SP}_S$ using a T_S-*good *order over the terms in the signature $\Sigma \cup \{\underline{a}, \underline{b}\}$ such that (i) every term over the subsignature $\Sigma_S^{\underline{a}}$ is smaller than any term that contains a symbol in $(\Sigma \setminus \Sigma_S) \cup \{\underline{b}\}$, (ii) not containing $\bot$, and such that (iii) s-rooted terms can be maximal just in ground equations in S_ω and (iv) variables of sort* NUM *are never the maximal term in the equations. The set $\Delta(\underline{a})$ of all the ground equations over $\Sigma_S^{\underline{a}}$ in S_ω is a T_S-basis for T.*

Proof.

Suppose that $T \cup T_S \cup G(\underline{a}, \underline{b}) \models l = r$, being $l = r$ a ground equation over $\Sigma_S^{\underline{a}}$. We want to show that already $T_S \cup \Delta(\underline{a}) \models l = r$.

A saturation of $Ax(T) \cup G(\underline{a}, \underline{b}) \cup \{l \neq r\}$ under $\mathcal{SP}_S$ is equal to a saturation of $S_\omega \cup \{l \neq r\}$. Since S_ω contains neither $\bot$, nor non-ground equations whose maximal term is s-rooted, nor equations whose maximal term is a variable of sort NUM, the only way to derive $\bot$ is by reducing $l \neq r$ via equations from $\Delta(\underline{a})$: indeed, $l \neq r$ is defined on the signature $\mathsf{s} \cup 0 \cup \underline{a}$ and, at this point, recalling also our choice of the reduction ordering, no equation in S_ω containing a symbol different from $\mathsf{s}, 0, \underline{a}$, i.e. no equation out of $\Delta(\underline{a})$, can be used to rewrite a term on signature $\mathsf{s}, 0, \underline{a}$.

Thus it follows that the saturation of $S_\omega \cup \{l \neq r\}$ will add only ground literals to S_ω, or $\bot$. In any case, the saturation still satisfies all the requirements in order to apply Theorem 1, and so we have the following chain of implications: $T \cup T_S \cup G(\underline{a}, \underline{b}) \models l = r$ iff the saturation of $Ax(T) \cup G(\underline{a}, \underline{b}) \cup \{l \neq r\}$ under $\mathcal{SP}_S$ contains $\bot$, iff saturation of $\Delta(\underline{a}) \cup \{l \neq r\}$ under $\mathcal{SP}_S$ contains $\bot$, iff $T_S \cup \Delta(\underline{a}) \models l = r$. The hypothesis that S_ω is finite guarantees that also $\Delta(\underline{a})$ is finite, i.e. $\Delta(\underline{a})$ is really a T_S-basis for T.

B Example of Data Structure: Trees with Size

For sake of completeness, we consider the theory of trees T_{BS} (defined in Section 3) which is not handled in [17]. We show that any saturation of a set of ground literals G and the axioms for T_{BS} under the rules of the calculus $\mathcal{SP}_S$ is finite, once chosen an appropriate T_S-*good* ordering.

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 and z are variables of sort TREES, t_1, t_2 are constants of sort TREES, e, e_1, e_2 are constants of sort ELEM, a, b are constants of sort NUM, r_1, r_2 stand for constants of sort NUM or terms of the kind $\mathsf{size}_L(t)$ or $\mathsf{size}_R(t)$ for some constant t of sort TREES, and the symbol $\bowtie$ is a shortening for both $=$ and $\neq$).

i.) equational axioms for trees

a) $\mathsf{size}_L(\mathsf{bin}(x, y, z)) = \mathsf{s}(\mathsf{size}_L(y))$;
b) $\mathsf{size}_L(\mathsf{null}) = 0$;
c) $\mathsf{size}_R(\mathsf{bin}(x, y, z)) = \mathsf{s}(\mathsf{size}_R(z))$;
d) $\mathsf{size}_R(\mathsf{null}) = 0$;

ii.) ground literals over the sort TREES

a) $\mathsf{bin}(e, t_1, t_2) = t_3$;
b) $t_1 \bowtie t_2$;

iii.) ground literals over the sort NUM

a) $r_1 = \mathsf{s}^m(r_2)$;
b) $\mathsf{s}^n(r_1) = r_2$;
c) $\mathsf{s}^n(a) \neq \mathsf{s}^m(b)$.

iv.) ground literals over the sort ELEM

a) $e_1 \bowtie e_2$;

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

- $\mathsf{bin} > t > \mathsf{null} > e > \mathsf{size}_L > \mathsf{size}_R$ for every constant t of sort TREES and every constant e of sort ELEM;
- $\mathsf{size}_L > \mathsf{size}_R > a > 0 > \mathsf{s}$ for every constant a of sort NUM;

These requirements over the precedence guarantee that every compound term of sort TREES is bigger than any constant and that $\succ$ is a *T_S-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).

Corollary 1. *For any set G of ground literals, any saturation of $Ax(T_{BS}) \cup G$ w.r.t. $\mathcal{SP}_S$ is finite.*

Proof. We can divide the literals above into the ground one and the non-ground. It is easy to check that, given the higher priority of the contraction rules, the set of the non-ground literals is saturated. Moreover, it is easy to verify that any saturation between non-ground literals and ground ones produces, whenever possible, literals that belong again to one of the group ii.), iii.) or iv.), and that are smaller then the ground literal used in the antecedent of the rule. Finally, the inferences between literals that are ground produce, naturally, ground literals that are smaller than the maximal literal in the antecedent of the inference.

Summing up, we have checked that each new literal that is derived during the saturation process is always ground and smaller in the ordering then the maximal ground literal in the antecedent of the rule used to produce it. Therefore, every literal produced during the saturation phase 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.

No meaningful variations appear when we consider the theory T'_{BS} in which the sort ELEM is identified with the sort NUM.

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