

Locality Transfer: From Constrained Axiomatizations to Reachability Predicates

Matthias Horbach and Viorica Sofronie-Stokkermans

University Koblenz-Landau and Max-Planck-Institut für Informatik Saarbrücken

Abstract. In this paper, we build upon our previous work in which we used constrained clauses in order to finitely represent infinite sets of clauses and proved that constrained axiomatizations are local if they are saturated under a version of resolution. We extend this result by identifying situations in which locality of saturated axiomatizations is maintained if we enrich the base theory by introducing new predicates (often reachability predicates) instead of using constraints for these properties.

1 Introduction

Many problems in computer science (e.g. in verification) can be reduced to checking the satisfiability of conjunctions of literals with respect to a theory. For efficient and accurate reasoning it is essential to reduce the search space without losing completeness and to make modular or hierarchical reasoning possible. In [13], we introduced the class of *local theory extensions* for which (i) complete instantiation schemes exist, and (ii) hierarchical and modular reasoning is possible. Locality is a property of an *axiomatization of a theory*; therefore it is very important to *recognize locality* of a set of clauses, and to *obtain local axiomatizations* by transforming non-local sets of clauses into local ones. In [2, 3], a link between (order)-locality and saturation under ordered (hyper)resolution is presented; this allows us to obtain, by saturation, local axiomatizations for a theory from non-local ones. Sometimes, however, the saturation process may not terminate. In [6] we showed that in order to obtain finite representations of possibly infinite sets of clauses we can use *constrained clauses*; we used a sound and complete ordered resolution and superposition calculus for constrained clauses and established a link between saturation in our calculus and order locality.

In spite of their advantages, axiomatizations using constrained clauses are difficult to export to standard first-order provers. In this paper we show that we can avoid using constrained clauses if we introduce new relations which encode the constraints. Our main contributions can be described as follows:

- We analyze possibilities of changing local sets of constrained clauses into local sets of clauses, by encoding the constraints using additional predicates.
- We identify situations in which the properties of the new predicates (which include reachability) can be encoded using first-order logic.
- We then identify situations in which locality results for the constrained clause set can be transferred to the new presentation defined this way.

Structure of the paper. The paper is structured as follows: In Sect. 2 we introduce the terminology we use and present the relevant results on local theory extensions. In Sect. 3 we identify a class of constrained clauses for which instantiation can be effectively computed. In Sect. 4 we analyze possibilities of changing local sets of constrained clauses into local sets of clauses, by encoding the constraints using additional predicates. In Sect. 5 we identify extensions of theories of absolutely free constructors for which locality is preserved after this transformation. Sect. 6 contains an overview of possible applications and further work.

2 Preliminaries

In this section we introduce the terminology and main results used in the paper.

2.1 General definitions

We build on the notions of [1, 7] and shortly recall here the most important concepts concerning terms and orderings and the specific extensions (concerning constrained clauses) needed in this article. For simplicity, we restrict to single-sorted signatures (the many-sorted case works similarly).

Terms and Clauses. Let $\Pi = (\Sigma, \text{Pred})$ be a *signature* consisting of a set Σ of function symbols of fixed arity and a set Pred of predicate symbols of fixed arity, and let X be a countably infinite set of variables such that X and Σ are disjoint. Terms s, t , equations $s \approx t$, atoms $P(\vec{t})$, atoms, literals, clauses and Horn clauses are defined as usual. We denote by $\mathsf{T}_\Sigma(X)$ the set of all terms over Σ and X and by T_Σ the set of all ground terms over Σ . To improve readability, term tuples $(t_1, \dots, t_n)$ will often be denoted by $\vec{t}$.

Substitution Expressions. A *substitution* σ is a map from a finite set $X' \subseteq X$ of variables to $\mathsf{T}_\Sigma(X)$. The application of σ to a term t or a term tuple $\vec{t}$ is denoted by $t\sigma$ or $\vec{t}\sigma$, respectively. *Substitution expressions* are built over substitutions and constructors $\circ$ (composition), $|$ (disjunction), and $*$ (loop) of arity 2, 2 and 1, respectively. Substitution expressions are denoted as $\bar{\sigma}, \bar{\tau}$. We will often write $\bar{\sigma} \circ \bar{\tau}$ as $\bar{\sigma}\bar{\tau}$ and $\bar{\sigma}\bar{\tau}^*$ as $\bar{\sigma}^+$. The *domain* $\text{dom}(\bar{\sigma})$ and the *variable range* $\text{VRan}(\bar{\sigma})$ of a substitution expression are defined as follows: For a substitution $\sigma : \{x_1, \dots, x_n\} \rightarrow \mathsf{T}_\Sigma(X)$, we define $\text{dom}(\sigma) = \{x_1, \dots, x_n\}$ and $\text{VRan}(\sigma) = \text{vars}(x_1\sigma, \dots, x_n\sigma)$. For complex substitution expressions, we have

$$\begin{aligned} \text{dom}(\bar{\sigma} \circ \bar{\tau}) &= \text{dom}(\bar{\sigma}) & \text{VRan}(\bar{\sigma} \circ \bar{\tau}) &= \text{VRan}(\bar{\tau}) \\ \text{dom}(\bar{\sigma}_1 | \bar{\sigma}_2) &= \text{dom}(\bar{\sigma}_1) \cup \text{dom}(\bar{\sigma}_2) & \text{VRan}(\bar{\sigma}_1 | \bar{\sigma}_2) &= \text{VRan}(\bar{\sigma}_1) \cap \text{VRan}(\bar{\sigma}_2) \\ \text{dom}(\bar{\sigma}^*) &= \text{dom}(\bar{\sigma}) & \text{VRan}(\bar{\sigma}^*) &= \text{dom}(\bar{\sigma}) \cup \text{VRan}(\bar{\sigma}) \end{aligned}$$

These notions are only intuitive for well-formed expressions: A substitution expression $\bar{\sigma}$ is *well-formed*, if (i) for each subexpression $\bar{\tau}_1 \circ \bar{\tau}_2$ of $\bar{\sigma}$, $\text{VRan}(\bar{\tau}_1) = \text{dom}(\bar{\tau}_2)$, (ii) for each subexpression $\bar{\tau}_1 | \bar{\tau}_2$, $\text{dom}(\bar{\tau}_1) = \text{dom}(\bar{\tau}_2)$ and $\text{VRan}(\bar{\tau}_1) = \text{VRan}(\bar{\tau}_2)$, and (iii) for each subexpression $\bar{\tau}^*$, $\text{VRan}(\bar{\tau}) = \text{dom}(\bar{\tau})$.

Constrained Clauses. A *constrained clause* $\alpha \parallel C$ consists of a clause C and a *regular constraint* α of the form $(x_1 \approx y_1, \dots, x_n \approx y_n)\bar{\sigma}$, also written as $(\vec{x} \approx \vec{y})\bar{\sigma}$,

such that x_i, y_i are variables and $\bar{\sigma}$ is a well-formed substitution expression with domain $\{x_1, y_1, \dots, x_n, y_n\}$. If a regular constraint α does not contain any equations, we call $\alpha \parallel C$ *unconstrained* and identify it with its clausal part C . If $\alpha = (\vec{x} \approx \vec{y})\bar{\tau}$ is a regular constraint, then $\alpha\sigma$ is defined as $(\vec{x} \approx \vec{y})\bar{\tau}\sigma'$, where $\sigma' : \text{VRan}(\bar{\tau}) \rightarrow \text{T}_\Sigma(X)$ maps z to $z\sigma$ if $z \in \text{dom}(\sigma)$ and to z otherwise. The application $(\alpha \parallel C)\sigma$ of a substitution to a constrained clause is then defined as $\alpha\sigma \parallel C\sigma$. The set of *ground instances* of a constrained clause $\alpha \parallel C$ consists of all ground clauses $C\sigma$ for which $\alpha\sigma$ is a satisfiable ground constraint. This means that regular constraints are interpreted syntactically. To improve readability of constraints, we sometimes use straightforward expansions of the substitution expressions. For example, $(x \approx y)\{x \mapsto x, y \mapsto s(y)\}^*$ might be written as $x \approx s^*(y)$.

Denotations and Models of Constrained Clauses. We define the *denotation* $\llbracket \sigma \rrbracket$ of a substitution expression $\bar{\sigma}$ inductively as follows:

$$\begin{aligned} \llbracket \sigma \rrbracket &= \{\sigma\} & \llbracket \bar{\sigma}\bar{\tau} \rrbracket &= \{\sigma\tau \mid \sigma \in \llbracket \bar{\sigma} \rrbracket, \tau \in \llbracket \bar{\tau} \rrbracket\} \\ \llbracket \bar{\sigma}_1 \bar{\sigma}_2 \rrbracket &= \llbracket \bar{\sigma}_1 \rrbracket \cup \llbracket \bar{\sigma}_2 \rrbracket & \llbracket \bar{\sigma}^* \rrbracket &= \bigcup_{n \geq 0} \llbracket \bar{\sigma}^n \rrbracket \end{aligned}$$

Here $\bar{\sigma}^0$ denotes the identity substitution on $\text{dom } \bar{\sigma}$ and $\bar{\sigma}^{n+1} = \bar{\sigma} \circ \bar{\sigma}^n$.

The semantics of the application of substitution expressions to terms and clauses and the semantics of constrained clause sets are defined just as one would expect by identifying a substitution expression with its denotation and by identifying a constrained clause $(\vec{x} \approx \vec{y})\bar{\sigma} \parallel C$ with the (potentially infinite) clause set $\{\vec{x}\sigma \approx \vec{y}\sigma \rightarrow C \mid \sigma \in \llbracket \bar{\sigma} \rrbracket\}$. Models and satisfiability of constrained clause sets are then defined straightforwardly using this set (cf. [7] for details).

Orderings. A (strict partial) *ordering* $\prec$ on a set S is a transitive and irreflexive binary relation on S . It is *total* if $s \prec t$ or $t \prec s$ whenever $s \neq t$. It is *well-founded* if there is no infinite descending chain $s_1 \succ s_2 \succ \dots$ of elements of S . A well-founded ordering $\prec$ on $\text{T}_\Sigma(X)$ is a *reduction ordering* if $t \prec t'$ implies $u[t\sigma] \prec u[t'\sigma]$ for all $t, t', u \in \text{T}_\Sigma(X)$ and all substitutions σ .

Let $\prec_\top$ be an ordering on $\text{T}_\Sigma(X)$ and let $\prec$ be an ordering on atoms over $\text{T}_\Sigma(X)$. Then $\prec$ is *compatible* with $\prec_\top$ if $A_1 \prec A_2$ whenever every term in A_1 is strictly bounded by a term in A_2 , i.e. if for each term t_1 in A_1 there is a term t_2 in A_2 such that $t_1 \prec_\top t_2$. Any ordering $\prec$ on atoms can be extended to clauses in a standard way (by setting $\neg A \succ A$ for all atoms A and taking the multiset extension of $\prec$). Let $\prec_\top$ be a term ordering and $\prec$ be an atom ordering. A ground clause is *reductive* (w.r.t. $\prec_\top$ and $\prec$) if all of its $\prec_\top$ -maximal terms appear in the $\prec$ -maximal atoms. A constrained clause is *reductive* (w.r.t. $\prec_\top$ and $\prec$) if all its ground instances are reductive (cf. [3]).

Inferences. A (clausal) *inference rule* is a relation on (constrained) clauses. Its elements are called *inferences* and written as

$$\frac{C_1 \dots C_k}{C} \quad \text{resp.} \quad \frac{\alpha_1 \parallel C_1 \dots \alpha_k \parallel C_k}{\alpha \parallel C}.$$

The clauses $C_1, \dots, C_k$ (resp. the constrained clauses $\alpha_1 \parallel C_1, \dots, \alpha_k \parallel C_k$) are called the *premises* and C ($\alpha \parallel C$) the *conclusion* of the inference. An *inference system* is a set of inference rules. In what follows we will use the standard

inference rules for ordered resolution and hyperresolution (with selection) (cf. e.g. [1]), as well extensions to constrained clauses (cf. [7]).

Redundancy. Let N be a set of clauses. A ground clause C is *redundant* w.r.t. N (and $\prec$) if it is entailed by ground instances of N which are smaller than C w.r.t. $\prec$. A constrained clause is *redundant* w.r.t. a constrained clause set N (and $\prec$) if all of its ground instances are redundant. An inference is *redundant* w.r.t. N if its conclusion is redundant w.r.t. N or if a premise C is redundant w.r.t. $N \setminus \{C\}$. A constrained clause set N is *saturated* (w.r.t. a given inference system) if each inference with premises in N is redundant w.r.t. N .

2.2 Local theories and theory extensions

Local theories. The notion of local set of Horn clauses (or local Horn theory) was introduced by Givan and McAllester in [5]. A *local set of Horn clauses* is a set of Horn clauses $\mathcal{K}$ such that, for any ground Horn clause C , $\mathcal{K} \models C$ only if already $\mathcal{K}[C] \models C$ (where $\mathcal{K}[C]$ is the set of instances of $\mathcal{K}$ in which all terms are subterms of ground terms in either $\mathcal{K}$ or C). Since the size of $\mathcal{K}[G]$ is polynomial in the size of G for a fixed $\mathcal{K}$ and satisfiability of sets of ground Horn clauses can be checked in linear time, the validity of ground Horn clauses w.r.t. local Horn theories can be checked in polynomial time. In [2, 3], Basin and Ganzinger defined *order locality*. Given a term ordering $\prec$, we say that a set $\mathcal{K}$ of clauses entails a ground clause C bounded by $\prec$ (notation: $\mathcal{K} \models_{\prec} C$), iff there is a proof of $\mathcal{K} \models C$ from those ground instances of clauses in $\mathcal{K}$ in which (under $\preceq$) each term is smaller than or equal to some term in C . A set of clauses $\mathcal{K}$ is *local with respect to $\prec$* if whenever $\mathcal{K} \models C$ for a ground clause C , then $\mathcal{K} \models_{\prec} C$.

Local theory extensions. In [13] the notion of locality for Horn clauses is extended to the notion of *local extension* of a base theory.

Let $\mathcal{T}_0$ be an arbitrary theory with signature $\Pi_0 = (\Sigma_0, \text{Pred})$, where the set of function symbols is Σ_0 . Let $\Pi = (\Sigma_0 \cup \Sigma, \text{Pred}) \supseteq \Pi_0$ be an extension by a non-empty set Σ of new function symbols and let $\mathcal{K}$ be a set of (implicitly universally closed) clauses in the extended signature. We will denote by Π^c the extension of Π with a fixed countable set of fresh constants. In what follows, we assume that all ground clauses we refer to contain symbols in Π^c . We say that an extension $\mathcal{T}_0 \cup \mathcal{K}$ of $\mathcal{T}_0$ is *local* if it satisfies the following condition¹:

- (Loc) For every set G of ground clauses in Π^c it holds that $\mathcal{T}_0 \cup \mathcal{K} \cup G \models \perp$ if and only if $\mathcal{T}_0 \cup \mathcal{K}[G] \cup G \models \perp$

where $\mathcal{K}[G]$ consists of those instances of $\mathcal{K}$ in which the terms starting with *extension functions* are in the set $\text{est}(\mathcal{K}, G)$ of extension ground terms (i.e. terms starting with a function in Σ) which already occur in G or $\mathcal{K}$. (Note that the variables in clauses in $\mathcal{K}$ which do not occur below extension functions will not be instantiated in $\mathcal{K}[G]$.) The notion of local theory extension generalizes the notion of *local theories*. In [9, 10] we generalized condition (Loc) by considering

¹ It is easy to check that the formulation we give here and that in [13] are equivalent.

operators on sets of ground terms. This allows us to be more flexible w.r.t. the instances needed. Let Ψ be a map associating with every set T of ground terms a set $\Psi(T)$ of ground terms. For any set G of (augmented) ground Π^c -clauses we write $\mathcal{K}[\Psi_{\mathcal{K}}(G)]$ for $\mathcal{K}[\Psi(\text{est}(\mathcal{K}, G))]$ (the set consists of those instances of $\mathcal{K}$ in which the terms starting with *extension functions* are in $\Psi(\text{est}(\mathcal{K}, G))$). We can define a version of locality (Loc^{Ψ}) in which the set of terms used in the instances of the axioms is described using the map Ψ .

(Loc^{Ψ}) For every set G of ground clauses in Π^c it holds that
 $\mathcal{T}_0 \cup \mathcal{K} \cup G \models \perp$ if and only if $\mathcal{T}_0 \cup \mathcal{K}[\Psi_{\mathcal{K}}(G)] \cup G \models \perp$.

Extensions satisfying condition (Loc^{Ψ}) are called Ψ -local. Local theory extensions are Ψ -local, where Ψ is the identity operator. The order-local theories introduced in [3] satisfy a $\Psi^{\preceq}$ -locality condition, where for every set T of ground clauses $\Psi^{\preceq}(T) = \{s \mid s \text{ ground term and } s \preceq t \text{ for some } t \in T\}$, where $\prec$ is the order on terms considered in [3].

Hierarchical reasoning. Let $\mathcal{T}_0 \subseteq \mathcal{T} = \mathcal{T}_0 \cup \mathcal{K}$ be a theory extension satisfying (Loc^{Ψ}). To check the satisfiability w.r.t. $\mathcal{T}$ of a formula G , where G is a set of ground Π^c -clauses, we proceed as follows:

Step 1: By locality, $\mathcal{T} \cup G \models \perp$ iff $\mathcal{T}_0 \cup \mathcal{K}[\Psi_{\mathcal{K}}(G)] \cup G \models \perp$.

Step 2: Purification. We purify $\mathcal{K}[\Psi_{\mathcal{K}}(G)] \cup G$ (by introducing, in a bottom-up manner, new constants c_t for subterms $t = f(g_1, \dots, g_n)$ with $f \in \Sigma$, g_i ground Π_0^c -terms, and corresponding definitions $c_t \approx t$) and obtain the set of formulae $\mathcal{K}_0 \cup G_0 \cup D$, where D consists of definitions $c_t \approx f(g_1, \dots, g_n)$, where $f \in \Sigma$, c_t is a constant, $g_1, \dots, g_n$ are ground Π_0^c -terms, and $\mathcal{K}_0, G_0$ are Π_0^c -formulae.

Step 3: Reduction to testing satisfiability in $\mathcal{T}_0$. We reduce the problem to testing satisfiability in $\mathcal{T}_0$ by replacing D with the following set of clauses:

$$\text{Con}_0 = \left\{ \bigwedge_{i=1}^n c_i \approx d_i \rightarrow c \approx d \mid f(c_1, \dots, c_n) \approx c, f(d_1, \dots, d_n) \approx d \in D \right\}.$$

This yields a sound and complete hierarchical reduction to a satisfiability problem in the base theory $\mathcal{T}_0$:

Theorem 1 ([9]) *Let $\mathcal{K}$ and G be as specified above. Assume that $\mathcal{T}_0 \subseteq \mathcal{T}_0 \cup \mathcal{K}$ satisfies condition (Loc^{Ψ}). Let $\mathcal{K}_0 \cup G_0 \cup \text{Con}_0$ be obtained from $\mathcal{K}[\Psi_{\mathcal{K}}(G)] \cup G$ by purification (cf. Step 2). Then $\mathcal{T}_0 \cup \mathcal{K} \cup G \models \perp$ if and only if $\mathcal{T}_0 \cup \mathcal{K}_0 \cup G_0 \cup \text{Con}_0 \models \perp$.*

Thus, satisfiability of ground clauses G as above w.r.t. $\mathcal{T}$ is decidable provided that $\Psi_{\mathcal{K}}(G)$ is effective, $\mathcal{K}[\Psi_{\mathcal{K}}(G)]$ is finite and $\mathcal{K}_0 \cup G_0 \cup \text{Con}_0$ belongs to a decidable fragment of $\mathcal{T}_0$.

Locality and embeddability. The (Ψ)-locality of an extension can be recognized by proving embeddability of partial models into total models assuming that the extension clauses are flat and linear; for details see [13, 9, 10]. The locality proofs given there also explain how to construct models of satisfiable ground (extended) clauses starting from models of their instances. We used the link between locality and embeddability for identifying various classes of local theory extensions. Some examples are given below:

Theories of absolutely free data structures. Let $\text{AbsFree}_{\Sigma_0} = (\bigcup_{c \in \Sigma_0} (\text{Injective}_c) \cup (\text{Acyclic}_c)) \cup \bigcup_{c,d \in \Sigma, c \neq d} \text{Disjoint}_{c,d}$, where:

$$\begin{array}{ll} \text{Injective}_c & c(x_1, \dots, x_n) \approx c(y_1, \dots, y_n) \rightarrow x_j \approx y_j \quad \text{for all } j \in \{1, \dots, n\}, \\ \text{Disjoint}_{c,d} & c(x_1, \dots, x_n) \not\approx d(y_1, \dots, y_m) \\ \text{Acyclic}_c & c(t_1, \dots, t_n) \not\approx x \text{ if } x \text{ occurs in some } t_i \quad (\text{an axiom schema}) \end{array}$$

Theorem 2 ([14]) *The following theories are local:*

- (a) *The theory $\text{AbsFree}_{\Sigma_0}$ of absolutely free constructors in Σ_0 .*
- (b) *$\mathcal{T}_{cs} = \text{AbsFree}_{\Sigma_0} \cup \text{Sel}(\Sigma_0)$, where $\text{Sel}(\Sigma_0) = \bigcup_{c \in \Sigma_0} \bigcup_{i=1}^{a(c)} \text{Sel}(sel_i^c, c)$ axiomatizes a family of selectors $sel_1^c, \dots, sel_n^c$, where $a(c)$ is the arity of c , corresponding to constructors $c \in \Sigma_0$, where:*

$$\text{Sel}(sel_i^c, c) \quad \forall x, x_1, \dots, x_{a(c)} \quad x \approx c(x_1, \dots, x_{a(c)}) \rightarrow sel_i^c(x) \approx x_i$$

The arguments in [14] can be used to prove locality if the axioms for selectors contain in addition to $\text{Sel}(sel_i^c, c)$ also the axiom:

$$\forall x, x_1, \dots, x_{a(d)} \quad x \approx d(x_1, \dots, x_{a(d)}) \rightarrow sel_i^c(x) \approx x \text{ if } d \neq c.$$

Locality and saturation. In [2, 3], Ganzinger and Basin established a link between saturation and order locality, and used these results for automated complexity analysis, and for obtaining local axiomatizations from non-local ones.

Theorem 3 ([2, 3]) *Let $\prec_\tau$ be a well-founded (possibly partial) term ordering and $\prec$ a compatible and total atom ordering. Let $\mathcal{K}$ be a set of clauses without equality which is reductive w.r.t. $\prec_\tau$ and $\prec$. If $\mathcal{K}$ is saturated w.r.t. $\prec$ -ordered resolution, then $\mathcal{K}$ is order local w.r.t. $\prec$.*

Example 4 *Theorem 3 is used in [4] for proving (by saturation) the locality of the presentation Int of the set of integers with successor and predecessor:*

$$\begin{array}{ll} (1) & p(x) \approx y \rightarrow s(y) \approx x \\ (2) & s(x) \approx y \rightarrow p(y) \approx x \end{array} \quad \begin{array}{ll} (3) & p(x) \approx p(y) \rightarrow y \approx x \\ (4) & s(x) \approx s(y) \rightarrow y \approx x \end{array}$$

together with an explicit axiomatization of the predicate $\approx$ by congruence axioms.

However, using saturation for detecting locality or for generating local presentations from non-local ones has the following drawbacks: (i) Equality cannot be used as a built-in predicate: If the clauses contain the equality predicate then the congruence axioms have to be added explicitly. (ii) The size of the saturated sets of clauses can be very large. Often, in fact, infinitely many clauses are generated.

Example 5 ([6]) *We illustrate the last problem by two examples:*

- (1) *From the clause set $\{x \leq x, x \leq y \wedge y \leq z \rightarrow x \leq z, f(x) \leq f(s(x))\}$ by saturation under ordered resolution we obtain an infinite set containing all clauses of the form $f(x) \leq f(s^n(x))$, where $n \geq 0$.*
- (2) *From the clause set $\{\text{nat}(0), \text{nat}(x), y \approx s(x) \rightarrow \text{nat}(y)\}$ by saturation under ordered resolution and equality factoring we obtain an infinite set containing all clauses of the form $\text{nat}(s^n(0))$.*

In such cases, a usual resolution-based theorem prover will not be able to detect saturation, since the set of clauses which are generated is infinite.

To reduce the size of a representation, we can use constrained clauses. In [6] we showed that Thm. 3 has a counterpart for constrained clauses with equality:

Theorem 6 ([6]) *Let $\prec_\tau$ be a reduction ordering and $\prec$ a compatible and total atom ordering. Let N be a set of constrained clauses that is reductive w.r.t. $\prec_\tau$ and saturated under the superposition calculus w.r.t. $\prec$ for constrained clauses described in [7]. Assume that each constrained clause in N with positive equational atoms contains either a unique positive equation which is also maximal, or a negative equation which is maximal. Let C be a ground clause whose antecedent does not contain any equations. Then $N \models C$ iff $N \models_{\leq} C$.*

The restrictions on N guarantee that constraint equations can be regarded syntactically and that all clauses that are derived in the constraint superposition calculus [7] from $N \cup \neg C$ contain equational atoms only negatively and no inferences into constraints are necessary.

The advantage of using constrained clauses is that in many cases it allows us to obtain a finite symbolic representation for possibly infinite sets of clauses.

Example 7 ([6]) *The constrained resolution calculus in [7] allows us to obtain a finite representation for the theory in Example 5(1) by using constrained clauses of the form $y \approx s^*(x) \parallel f(x) \leq f(y)$. For the theory described in Example 5(2), we obtain the saturated set of constrained clauses $\{x \approx s^*(0) \parallel \text{nat}(x)\}$.*

3 Enumerating ground instances of constrained clauses

Locality of theories and theory extensions guarantees that in order to disprove a set G of ground clauses we only need to look at certain ground instances of the local axiomatization (without loss of completeness). Note that the constraints are used as a notation for encoding a family of (syntactic) instances of a clause. The same holds when we compute given instances of such constrained clauses. We have $(\alpha \parallel C)[\Psi(G)] \subseteq C[\Psi(G)]$. A ground instance $C\tau$ from $C[\Psi(G)]$ is in $(\alpha \parallel C)[\Psi(G)]$ if the constraint $\alpha\tau$ holds. Unfortunately, it is in general not possible to effectively enumerate all such instances. This is due to the fact that satisfiability of constraints is not decidable (cf. [7]).

We show that enumerating ground instances is possible for an important class of constraints; we introduce a notation for them below:

Notation 8 *The expression $\vec{x} \approx \vec{y} \bar{\sigma}$ denotes a constraint $(\vec{x} \approx \vec{y}) \bar{\sigma}$ where (i) $\bar{\sigma}$ is the identity on $\vec{x}$ and (ii) no new occurrences of variables in $\vec{x}$ are introduced by $\bar{\sigma}$, i.e. $\vec{x} \cap \text{vars}(\vec{y} \bar{\sigma}) = \emptyset$.*

In our experience, such constraints appear very frequently. Examples are the local axiomatization of monotonicity in Example 7, as well as the following axiomatization of the theory of absolutely free constructors $c_1, \dots, c_m$ with arities $n_1, \dots, n_m$ (which satisfies the requirements of Thm. 6):

- Injectivity: $c_i(x_1, \dots, x_n) \approx c_i(y_1, \dots, y_n) \rightarrow x_j \approx y_j$ for all $j \in \{1, \dots, n\}$;
- Disjointness: $c_i(x_1, \dots, x_{n_i}) \not\approx c_j(y_1, \dots, y_{n_j})$ for $i < j$;
- Acyclicity: $x \approx y(\sigma_{11} | \dots | \sigma_{mn_m})^+ \parallel x \not\approx y$, where $\sigma_{ij} = \{y \mapsto c_i(\dots, y, \dots)\}$ with y at position j and fresh variables at all other positions.

Theorem 9 *For every set N of constrained clauses with only constraints of the form $\vec{x} \approx \vec{y} \vec{\sigma}$ and every set T of ground terms we can effectively enumerate $N[T]$.*

Proof: A ground instance $C\tau$ of $C[T]$ is in $(\alpha \parallel C)[T]$ if after applying the substitution τ the constraint holds. Whether this is the case can be decided for constraints of the form $\vec{x} \approx \vec{y} \vec{\sigma}$ by solving the matching problem $\vec{x}\tau = \vec{y}\vec{\sigma}\tau$, where $\vec{x}\tau$ consists of ground terms only. This is done by a reduction in the style of the standard unification algorithm by Martelli and Montanari, recursively unfolding the substitution expression:

$$\begin{aligned}
(\vec{s}_1, f(\vec{s}_2)) \approx (\vec{t}_1, g(\vec{t}_2)) \vec{\sigma} &\rightsquigarrow \perp & \vec{s} \approx \vec{y} \vec{\sigma} \vec{\tau} &\rightsquigarrow \vec{s} \approx \vec{t} \vec{\tau} \text{ if } \vec{y} \vec{\sigma} = \vec{t} \\
(\vec{s}_1, f(\vec{s}_2)) \approx (\vec{t}_1, f(\vec{t}_2)) \vec{\sigma} &\rightsquigarrow (\vec{s}_1, \vec{s}_2) \approx (\vec{t}_1, \vec{t}_2) \vec{\sigma} & \vec{s} \approx \vec{y} \vec{\sigma}^* \vec{\tau} &\rightsquigarrow \vec{s} \approx \vec{y}(\vec{\tau} | \vec{\sigma}^* \vec{\tau}) \\
\vec{s} \approx \vec{y}(\vec{\sigma}_1 | \vec{\sigma}_2) \vec{\tau} &\rightsquigarrow \vec{s} \approx \vec{y} \vec{\sigma}_1 \vec{\tau} \mid \vec{s} \approx \vec{y} \vec{\sigma}_2 \vec{\tau} & (s, \vec{t}) \approx (x, \vec{y}) &\rightsquigarrow \vec{t} \approx \vec{y} \{x \mapsto s\}
\end{aligned}$$

We split the derivation into two branches for disjunctions. If at least one of the branches terminates without yielding $\perp$, the constraint is satisfiable. $\square$

Example 10 *Consider the constrained clause $(x \approx y) \sigma^* \parallel f(y) \leq f(x)$ with $\sigma = \{x \mapsto x, y \mapsto s(y)\}$, which axiomatizes monotonicity of a function f over the integers. Let $G = (d \approx s^3(c) \wedge f(c) > f(d))$. Then $\Psi(G) = \{d, c, s(c), s^2(c), s^3(c)\}$ and $((x \approx y) \sigma^* \parallel f(y) \leq f(x))[\Psi(G)]$ consists of the clauses:*

$$f(c) \leq f(s^i(c)) \text{ for } i \in \{0, 1, 2, 3\} \quad \text{and} \quad f(d) \leq f(d).$$

4 Enriching the language

Axioms consisting of constrained clauses are difficult to export to standard first-order provers. In addition, there are limitations in using them if we are interested in checking satisfiability in Herbrand models of the axioms, in which all Skolem constants of G are interpreted as ground terms.

Example 11 *Let $N = \{x \approx s^*(0) \parallel \text{nat}(x)\}$. If a is a constant, we cannot directly express the positive fact that there is an n such that $a = s^n(0)$, so we cannot prove $\text{nat}(x) \rightarrow \text{nat}(s(x))$ from N : For the ground clause set $G = \{\text{nat}(a), \neg \text{nat}(s(a))\}$, we cannot derive a contradiction using the axiomatization N . (In fact, $N \cup G$ has a model with the universe $\{s^n(0) \mid n \in \mathbb{N}\} \cup \{s^n(a) \mid n \in \mathbb{N}\}$.)*

To address these limitations, and to obtain more natural axiomatizations, we *enrich the language with additional predicates* encoding the constraints and analyze situations in which locality results can be obtained for these new presentations.

Let N be a set of clauses with constraints of the form $\vec{x} \approx \vec{y} \vec{\sigma}$ satisfying the conditions from Thm. 6. Assume that N is saturated (hence local). Our goal is to use the form of the constraints to (conservatively) extend the language – e.g.

by defining new predicates – in order to obtain local presentations expressible as sets of clauses without constraints. For this we extend the signature by adding a new predicate symbol $R_{\bar{\sigma}}$ for every constraint $\alpha = \bar{x} \approx \bar{y} \bar{\sigma}$ and rewriting every constrained clause of the form $\alpha \parallel C$ into the clause $R_{\bar{\sigma}}(\bar{x}, \bar{y}) \rightarrow C$. We can encode the properties of these new relations (i) by giving a complete description for all ground terms, or (ii) by providing axiomatizations.

4.1. Concrete description of the new relations. In a first step, we assume that we give a complete description of the newly introduced relations for all ground terms; we will have to ensure that saturation is preserved when changing the clauses in N and adding the encoding. For this it is necessary that, whenever we eliminate the constraint from a constrained clause $\alpha \parallel C$, the ground literal $\neg R(\bar{x}, \bar{y})$ in the resulting clause $R(\bar{x}, \bar{y}) \rightarrow C$ does not influence the size of that clause too much. This can be achieved by ignoring constraint equations $x \approx y$ where x does not appear in the clausal part.

Example 12 *The clause $x \approx s(y) \parallel P(y)$ is not translated to $R(x, y) \rightarrow P(y)$, where the R literal dominates all ground instances of the form $R(s(t), t) \rightarrow P(t)$, but to $R() \rightarrow P(y)$, where R is the predicate corresponding to the empty constraint, or equivalently to $\rightarrow P(y)$.*

We show that if N is saturated under superposition for constraint clauses, this holds for the clause sets where constraints have been replaced by new predicates.

Theorem 13 *Let N be a consistent set of constrained clauses over $\Pi = (\Sigma, \text{Pred})$ with constraints of the form $\bar{x} \approx \bar{y} \bar{\sigma}$. Let N' be obtained from N by replacing the constraints by fresh literals as explained above. Let $\mathcal{R}$ be the set of ground instances of these literals, such that $R_{\bar{\sigma}}(\bar{x}, \bar{y})\tau \in \mathcal{R}$ if $(\bar{x} \approx \bar{y})\bar{\sigma}\tau$ is valid, and $\neg R_{\bar{\sigma}}(\bar{x}, \bar{y})\tau \in \mathcal{R}$ otherwise. Let $\prec_{\top}$ be a reduction ordering and $\prec$ a compatible and total atom ordering which ensures that R -literals are not maximal in any clause in N' . If N is (i) saturated under ordered resolution w.r.t. $\prec$, or (ii) saturated under constrained superposition w.r.t. $\prec$, or (iii) Horn and peak saturated under ordered hyperresolution w.r.t. $\prec$, then the same holds for $N' \cup \mathcal{R}$.*

It can be seen that for every set of ground clauses G in the signature of N , $N \cup G$ is satisfiable iff $N' \cup \mathcal{R} \cup G$ is satisfiable. By Thm. 13, if N is a set of constrained clauses in first-order logic without equality which is saturated under ordered resolution, or it satisfies the assumptions in Thm. 6 and is saturated under constrained superposition, so $N' \cup \mathcal{R} \cup G \models \perp$ iff $(N' \cup \mathcal{R})[\Psi^{\prec}(G)] \cup G \models \perp$.

4.2. Axiomatization for the newly introduced predicates. There are some problems with the approach described above. Assume that G contains constants which are not in Σ . Then, instances of the constraints $\alpha = \bar{x} \approx \bar{y} \bar{\sigma}$ in N containing such constants might be true for some interpretations of the constants and false for other interpretations. In such a situation we cannot add the corresponding instances of the $R_{\bar{\sigma}}$ -literals to $\mathcal{R}$. If σ is a substitution expression we need to express the properties of $R_{\bar{\sigma}}$ in terms of the substitutions used in σ . For this, we introduce for every constraint of the form $\bar{x} \approx \bar{y} \bar{\sigma}$ a set of first-order formulae $\text{Def}_{\bar{\sigma}}$ describing the relation $R_{\bar{\sigma}}(\bar{x}, \bar{y})$ by induction on the structure of α as follows:

Definition 14 (Axiomatizing the properties of the new relations)

- (1) For atomic substitutions σ : $\text{Def}_\sigma := R_\sigma(\vec{x}, \vec{y}) \leftrightarrow (x_1 \approx \sigma(y_1) \wedge \dots \wedge x_n \approx \sigma(y_n))$
- (2) For compositions: $\text{Def}_{\bar{\sigma}\bar{\tau}} := R_{\bar{\sigma}\bar{\tau}}(\vec{x}, \vec{y}) \leftrightarrow \exists \vec{z} (R_{\bar{\sigma}}(\vec{x}, \vec{z}) \wedge R_{\bar{\tau}}(\vec{z}, \vec{y}))$
- (3) For disjunctions: $\text{Def}_{\bar{\sigma}_1 | \bar{\sigma}_2} := R_{\bar{\sigma}_1 | \bar{\sigma}_2}(\vec{x}, \vec{y}) \leftrightarrow (R_{\bar{\sigma}_1}(\vec{x}, \vec{y}) \vee R_{\bar{\sigma}_2}(\vec{x}, \vec{y}))$
- (4) For iterations: $\text{Def}_{\bar{\sigma}^*}$, axiom expressing the fact that $R_{\bar{\sigma}^*}$ is the reflexive and transitive closure of $R_{\bar{\sigma}}$.

Example 15 Let σ be a substitution with $\sigma(y) = s(y)$ and $\sigma(x) = x$. Let τ be the substitution with $\tau(y) = s(s(y))$ and $\tau(x) = x$. Let $\alpha = x \approx y \sigma$ (which corresponds to $x \approx s(y)$), and $\beta = x \approx y \tau$ (which corresponds to $x \approx s(s(y))$). Then $\text{Def}_\sigma: R_\sigma(x, y) \leftrightarrow (x \approx s(y))$; $\text{Def}_\tau: R_\tau(x, y) \leftrightarrow (x \approx s(s(y)))$; $\text{Def}_{\sigma\tau}: R_{\sigma\tau}(x, y) \leftrightarrow \exists z (R_\sigma(x, z) \wedge R_\tau(z, y))$; $\text{Def}_{\sigma|\tau}: R_{\sigma|\tau}(x, y) \leftrightarrow (R_\sigma(x, y) \vee R_\tau(x, y))$.

It can be shown that if N is a set of constrained clauses and G is a set of ground clauses, $N \cup G$ is satisfiable iff $N' \cup G$ has a model satisfying definitions (1)-(3) in Def. 14 and such that for every substitution expression $\bar{\sigma}$, $R_{\bar{\sigma}^*}$ is the reflexive and transitive closure of $R_{\bar{\sigma}}$. The definitions of relations $R_{\bar{\sigma}}$ where $\bar{\sigma}$ is a composition or a disjunction of substitutions can be expressed using first-order formulae over relations associated with the component substitutions of $\bar{\sigma}$. The problematic part is finding a set of axioms which expresses the fact that $R_{\bar{\sigma}^*} = (R_{\bar{\sigma}})^*$. It is known that transitive closure can be encoded in first-order logic (in an extension of the original signature) if we are interested only in finite models. An axiomatization for reflexive transitive closure which is sound and complete for finite domains was proposed in [12]; an axiomatization which completely describes transitive closure over finite, acyclic graphs was given in [11]:

$$T_1(R) : \forall x, y (R^*(x, y) \leftrightarrow (x \approx y \vee \exists z (R(x, z) \wedge R^*(z, y)))).$$

In [11] it is shown that for checking validity of formulae which contain predicates of the form R^* only positively, only the $\leftarrow$ implication $T_1^{\leftarrow}(R)$ of $T_1(R)$ is needed. Having a first order axiomatization does not mean that we have decidability (see the discussions on this in [11]). Consider for instance the axiom $T_2(R)$:

$$T_2(R) : \forall x, y (R^*(x, y) \leftrightarrow (x \approx y \vee \exists z (R^*(x, z) \wedge R(z, y)))).$$

Neither $T_2(R)$ nor $T_1(R)$ can be proved from the other without induction; similarly for the transitivity of the congruence closure [11].

5 Extensions of theories of absolutely free constructors

To identify situations in which axiomatizations with good properties exist, we make the following assumptions:

- (A1) We consider sets $N = N_0 \cup N_1$ of constrained clauses where:
 - N_0 is a set of axioms for $\mathcal{T}_{cs}$, e.g. $\text{AbsFree}_{\Sigma_0} \cup \text{Sel}(\Sigma_0)$ (cf. Thm. 2, [14]);
 - the extension $N_0 \subseteq N = N_0 \cup N_1$ is a local theory extension with the property that in N_1 every variable occurs below an extension function and every weak partial model can be extended to a total model with the same universes for the base sorts. (This happens for instance in the case of extensions with function symbols whose arguments are of base sorts, and where the codomain is of a new sort.)

(A2) The clauses in N contain only constraints of the form $\vec{x} \approx \vec{y} \vec{\sigma}$; where σ used only constructors, and $\prec$ is a simplification ordering.

We analyze the special situation in which the properties of the new relation symbols can be described by a set of formulae Def (of the form (1)-(4) in Def. 14) which can be expressed as a set of clauses without additional Skolem functions. This is the case, for instance, when the following conditions hold:

- (C1) Only composition of basic substitutions is allowed; iteration is only allowed for substitutions with $\sigma(x) \succeq x$ for every variable x .
- (C2) The set G of clauses for which we want to check satisfiability contains only negative atoms starting with a relation of form R_{σ^*} .

We first analyze satisfiability w.r.t. absolutely free models, possibly with additional generators (5.1) and then satisfiability in the absolutely free (Herbrand) model over Σ_0 , with no additional generators (5.2).

5.1. Satisfiability w.r.t. absolutely free models. We first show that under the assumptions above locality is preserved if we replace the constraints in the clauses using new predicate symbols and we specify their properties.

Theorem 16 *Let $N_1, \mathcal{R}, N'_1$ be sets of clauses as in Theorem 13. Assume that conditions (A1), (A2) and (C1) hold for $N = N_0 \cup N_1$. Let G be a set of ground clauses satisfying condition (C2). Then the following are equivalent:*

- (1) $\mathcal{R} \cup N_0 \cup N'_1 \cup G$ has a model A in which the interpretations of the new relations satisfy definitions (1)–(3), and such that $R_{\sigma^*}^A = (R_{\sigma}^A)^*$.
- (2) $(\text{Def} \cup N_0 \cup N'_1)[\Psi^{\preceq}(G)] \cup G$ has a finite partial model (having as elements the finite set of interpretations of terms in $\Psi^{\preceq}(G)$), where Def consists of all definitions of the form (1)–(4) in Def. 14 (which are in this case clauses without additional Skolem functions), and where the definitions of type (4) are axiomatized by the set of clauses consisting for every $R_{\bar{\sigma}^*}$ of $T_1^{\leftarrow}(R_{\bar{\sigma}})$.

Example 17 *Let $N_f = \{x \approx s^*(y) \mid f(y) \leq f(x)\} \cup \text{Pre}$ be the (local) constrained axiomatization of the theory of monotonicity for the function f , where Pre contains the reflexivity and transitivity axioms for $\leq$. Let N_0 be the axiomatization of the absolutely free constructor s . Note that $x \approx s^*(y)$ is a shorthand for $x \approx y \sigma^*$, where $\sigma(x) = x$ and $\sigma(y) = s(y)$. The set of clauses obtained from N_f by replacing the constraints with new literals is $N'_f = \{R_{\sigma^*}(x, y) \rightarrow f(y) \leq f(x)\} \cup \text{Pre}$. For defining R_{σ} we use the axiom $\text{Def}_{\sigma} := \{R_{\sigma}(x, y) \leftrightarrow x \approx s(y)\}$. Let Def_{σ^*} be an axiomatization for R_{σ^*} as R_{σ}^* (e.g. as in [11]).*

Let $G_1 := a \approx s(b) \wedge \neg R_{\sigma^}(a, b) \wedge \neg f(b) \leq f(a)$ (containing only negative R_{σ^*} -literals). The set of clauses $N_0 \cup N_f \cup G_1$ is satisfiable if and only if $N_0 \cup N'_f \cup \mathcal{R} \cup G_1$ is satisfiable. By Thm. 16 this happens if and only if $(\text{Def} \wedge N'_f)[\Psi^{\preceq}(G_1)] \cup G_1$ is satisfiable, where $\text{Def} = \text{Def}_{\sigma} \wedge T_1^{\leftarrow}(R_{\sigma})$. We check the satisfiability of this set of ground clauses using the hierarchical reduction in Theorem 1.*

D	$G_1^c \cup N_0[\Psi^{\preceq}(G)]_0$	$G_1^{\mathcal{R}} \cup \text{Def}[\Psi^{\preceq}(G)]_0$	G_1^e	$N'_f[\Psi^{\preceq}(G)]_0$
$a_1 \approx f(a)$	$a \approx s(b)$	$\neg R_{\sigma^*}(a, b)$	$\neg a_1 \leq b_1$	$a_1 \leq a_1$ $R_{\sigma^*}(a, b) \rightarrow a_1 \leq b_1$
$b_1 \approx f(b)$	$\neg a \approx s(a)$	$\text{Def}_{\sigma}[\Psi^{\preceq}(G)]$		$b_1 \leq b_1$ $R_{\sigma^*}(b, a) \rightarrow b_1 \leq a_1$
	$\neg b \approx s(b)$	$T_1^{\leftarrow}(R_{\sigma})[\Psi^{\preceq}(G)]$		...

Let $G_2 := (b \approx s(a) \wedge R_{\sigma^*}(c, b) \wedge d \approx s(c) \wedge \neg f(a) \leq f(d))$. G_2 contains positive R_{σ^*} -literals, so Theorem 16 cannot be used. We present a proof of unsatisfiability of $\text{Def}_{\sigma} \cup T_1(R_{\sigma}) \cup T_2(R_{\sigma}) \cup N_f' \cup G_2$: From $b \approx s(a)$ and $d \approx s(c)$ and from Def_{σ} it follows that $R_{\sigma}(b, a)$ and $R_{\sigma}(d, c)$. From $R_{\sigma}(d, c)$, $R_{\sigma^*}(c, b)$ and $T_1(R_{\sigma})$ it follows that $R_{\sigma^*}(d, b)$. From $R_{\sigma^*}(d, b)$, $R_{\sigma}(b, a)$ and $T_2(R_{\sigma})$ it follows that $R_{\sigma^*}(d, a)$. From this and $N_f[\Psi^{\preceq}(G)]_0$ we derive $a_1 \leq d_1$, which together with $\neg a_1 \leq d_1$ leads to a contradiction.

5.2. Satisfiability w.r.t. Herbrand models over Σ_0 . Let $\mathcal{T}_{cs} := \text{AbsFree}_{\Sigma_0} \cup \text{Sel}(\Sigma_0)$ be the theory of absolutely free constructors in a finite set Σ_0 with corresponding selectors mentioned in Theorem 2. We analyze the problem of checking satisfiability of ground clauses w.r.t. such axiomatizations in *Herbrand models having as universe the set T_{Σ_0} of ground terms over the signature of $\mathcal{T}_{cs}$ (and no additional generators)*. Since the set T_{Σ_0} is infinite, we cannot use the results on first-order definability for transitive closure mentioned before. We will overcome the problem by noting the following fact:

Lemma 18 *For every element t in the absolutely free algebra T_{Σ_0} , the number of terms in the set $T = \bigcup_{i \in \mathbb{N}} T_i$, where $T_0 = \{t\}$ and $T_{i+1} = \{\text{sel}_j^c(t') \mid \text{sel}_j^c \text{ selector and } t' \in T_i\}$ is finite.*

Example 19 (i) *For the theory of one unary constructor s with selector p and one constant constructor 0 , T_{Σ_0} is the set of natural numbers; for every natural number m the set of elements of the form $p^n(m)$ with $n \in \mathbb{N}$ is finite. (ii) *For the theory of one binary constructor cons with selector tail and one constant constructor nil (modeling the theory of finite, acyclic lists), from every node in a linked, finite acyclic list, nil can be reached in finitely many steps using tail .**

It is known that $\mathsf{T}_{\Sigma_0} \models t \approx c(t_1, \dots, t_n) \leftrightarrow \bigwedge_{i=1}^n \text{sel}_i^c(t) \approx t_i \wedge \bigwedge_{i=1}^n t \not\approx t_i$. We can transform all clauses and all constraints using this equivalence, such that all constructors are eliminated. (The constrained clause $x \approx s^*(y) \parallel f(y) \leq f(x)$ can be rewritten to $y \approx p^*(x) \parallel f(y) \leq f(x)$.)

For the sake of simplicity we restrict here to signatures which contain one constructor with arity n and selectors $\text{sel}_1, \dots, \text{sel}_n$ and m constructors $c_1, \dots, c_m$ with arity 0. Let $\mathcal{T}_{sel}$ be axiomatized by the following axioms:

$$\begin{aligned} (S1) \quad & \bigwedge_{i=1}^n \text{sel}_i(x) \approx \text{sel}_i(y) \rightarrow x \approx y \vee (\bigvee_{i=1}^m x \approx c_i \vee \bigvee_{i=1}^m y \approx c_i) \\ (S2) \quad & \text{sel}_i(x) \approx x \rightarrow \bigvee_{i=1}^n x \approx c_i \end{aligned}$$

In order to ensure that we check ground satisfiability over T_{Σ_0} , we will add the following axiom (after extending the signature to a signature $\Sigma_{\mathcal{R}}$ by adding the relations of the form $R_{\bar{\sigma}}$):

$$(\text{Fin}) \quad R_{(\text{sel}_1 | \dots | \text{sel}_k)^*}(c_1^0, x) \vee \dots \vee R_{(\text{sel}_1 | \dots | \text{sel}_k)^*}(c_m^0, x)$$

where $\text{sel}_1, \dots, \text{sel}_k$ are all selectors and $c_1^0, \dots, c_m^0$ are all 0-ary constructors.²

² This is different from the approach used in [7] in which satisfiability is checked in the minimal Herbrand model; to ensure that all constants can be expressed using the initial signature, existential variables are added to the constraints.

Theorem 20 *Let A be a model of an extension of $\mathcal{T}_{cs}$ with additional relations $\{R_{\bar{\sigma}} \mid \bar{\sigma} \text{ substitution expression occurring in } \mathcal{K}\}$, where $\mathcal{K}$ is a family of constrained clauses satisfying conditions (1)-(4). Assume that Fin holds in A . Then A has as universe the set T_{Σ_0} .*

Having first order axiomatizations for relations of the form $R_{\bar{\sigma}^*}$ and transitive closure of $R_{\bar{\sigma}}$ does not necessarily imply that we have already methods for efficiently reasoning in such theories. We show that under assumptions (A1), (A2), (C1) and (C2), when restricting to models with universe T_{Σ_0} , checking satisfiability of sets of ground clauses G w.r.t. a set of saturated constrained clauses $N_0 \cup N_1$, where N_0 is a set of axioms for $\mathcal{T}_{cs}$, can be reduced to checking satisfiability of $N_0 \cup N'_1 \cup \text{Def} \cup G$ (where N'_1 is obtained from N_1 by replacing every clause $\bar{x} \approx \bar{y} \bar{\sigma} \parallel C$ in N with $R_{\bar{\sigma}}(\bar{x}, \bar{y}) \rightarrow C$ and Def is a set of clauses specifying definitions for the newly introduced relations) w.r.t. certain finite structures with universe T_G , defined by:

$$T_G = \bigcup_{i \in \mathbb{N}} T_i, \text{ where } T_0 = \text{st}(G) \text{ and } T_{i+1} = \{\text{sel}_i(t) \mid t \in T_i\}.$$

Theorem 21 *Let N_1 be a set of constrained clauses. Assume that conditions (A1), (A2) hold for $N = N_0 \cup N_1$. Assume that the definitions of the newly introduced predicate symbols can be expressed as a set Def of clauses without additional Skolem functions. For every set G of ground clauses satisfying condition (C2), the following are equivalent:*

- (1) $N_0 \cup N_1 \cup G$ has a model A with support T_{Σ_0} in which all relations satisfy conditions (1)-(3) in Def. 14 and for every σ , $R_{\sigma^*}^A = (R_{\sigma}^A)^*$.
- (2) $(\mathcal{T}_{\text{sel}} \cup \text{Def} \cup N'_1)[T_G] \cup G$ has a finite model P having as universe a subset of T_{Σ} containing all 0-ary constructors and with the property that all terms in T_G are defined in P .

Sometimes the assumptions can be relaxed. One of the simpler cases studied in the literature are theories with relations defined by $R(x, y) \leftrightarrow y \approx f(x)$, where f is a unary function symbol. We will therefore restrict here to analyzing the situation in which we have one unary constructor and several 0-ary constructors. For this case we use an adaptation of the axiomatization of finite, acyclic lists with reachability proposed in [16]. For the theory of one binary constructor, the results in [15] can be adapted; we think that the results in [15] can be used, with small changes, also for the theory of one n -ary constructor.

Example 22 *Let R_{p^*} be the transitive closure of R_p , where $R_p(x, y) \leftrightarrow x \approx p(y)$. We adapt the axiomatization used in [16] to obtain an axiomatization Def_{p^*} for R_{p^*} consisting in the formulae (1)-(9). We included also the clause $\text{Fin} : (R_{p^*}(0, x))$, expressing the fact that from every element we can reach 0 in finitely many steps; this guarantees that we check satisfiability in the initial model:*

- | | |
|---|---|
| (1) $R_{p^*}(0, x)$ | (2) $p(x) \approx p(y) \rightarrow x \approx y \vee x \approx 0 \vee y \approx 0$ |
| (3) $R_{p^*}(x, x)$ | (4) $R_p(x, y) \rightarrow R_{p^*}(x, y)$ |
| (5) $p(x) \approx x \rightarrow x \approx 0$ | (6) $R_{p^*}(x, y) \rightarrow x \approx y \vee R_{p^*}(x, p(y))$ |
| (7) $R_{p^*}(x, y) \wedge R_{p^*}(y, x) \rightarrow x \approx y$ | (8) $R_{p^*}(x, y) \wedge R_{p^*}(y, z) \rightarrow R_{p^*}(x, z)$ |
| (9) $R_{p^*}(x, y) \wedge R_{p^*}(x, z) \rightarrow R_{p^*}(y, z) \vee R_{p^*}(z, y)$ | |

The locality proof in [16] uses the link between locality and embeddability; it can be adapted to prove that $\mathcal{T}_{sel} \cup \text{Def}_{p^*}$ is a local theory, or alternatively that the extension $N_1 \subseteq \mathcal{T}_{sel} \cup \text{Def}_{p^*}$ is local, where N_1 consists of the axioms (1), (3), (8) and (9) in Def_{p^*} .

Example 23 Consider the saturated axiomatization $N = \{x \approx s^*(0) \parallel \text{nat}(x)\}$ of the natural numbers. We transform N by using the selector instead and associating new predicate symbols with the constraints as follows: Since $x \approx s^*(0)$ iff $p^*(x) \approx 0$, we introduce new predicates R_σ and R_{σ^*} axiomatized by Def_σ : $R_\sigma(x, y) \leftrightarrow p(x) \approx y$; the axiomatization for R_{σ^*} is the one in Example 22. Then $N' := R_{\sigma^*}(x, 0) \rightarrow \text{nat}(x)$. Let Fin be $R_{\sigma^*}(x, 0)$.

Let $G = \{\text{nat}(a), \neg \text{nat}(s(a))\}$. By Thm. 21 we know that if $\mathcal{T}_s$ is the theory of one unary constructor s and one 0-ary constructor 0 , then there is a model of $N' \cup G \cup \text{Fin}$ (which has as support T_s , i.e. the set of natural numbers) if and only if $(\mathcal{T}_{sel} \cup N' \cup \text{Fin})[T_G] \cup G$ has a model. Note that although T_G is finite, we cannot estimate its size. We will start by instantiating the clauses in $\mathcal{T}_{sel} \cup N' \cup \text{Fin}$ using terms occurring in G . From $\neg \text{nat}(s(a))$ we derive $\neg R_{\sigma^*}(s(a), 0)$, which together with Fin leads to a contradiction. We do not need other instances.

6 Conclusions and future work

This paper continues our previous work [6], where we used constrained clauses to finitely represent possibly infinite sets of clauses and showed that constrained axiomatizations are local if they are saturated under a version of resolution. Here we showed that we can encode the constraints by extending the signature with new predicates – which often bear a particularly useful semantics – in such a way that locality is guaranteed. We focused on a special type of constrained clauses containing only constraints of the form $\vec{x} \approx \vec{y}\vec{\sigma}$, where $\vec{x} \cap \text{vars}(\vec{y}\vec{\sigma}) = \emptyset$ and identified extensions of theories of absolutely free constructors for which locality is preserved after this transformation.

Beyond what we presented in this paper, there are other ways of using such language extensions in more general settings, which we would like to explore in future work. Of particular interest are theories with fixpoints. Consider for instance the theory of a monotone function f over a theory with an underlying complete $\wedge$ -semilattice structure with top element 1. We can use the fact that greatest fixpoints exist to replace constraints of the form $x \approx f^*(1) \parallel c \leq x$ with the unconstrained clause $c \leq \text{gfp}(f)$. In future work we plan to apply this type of reasoning to the description logics $\mathcal{EL}$ or $\mathcal{ALC}$ with fixpoints (e.g. for computing uniform interpolants). Another promising research direction is inductive theorem proving: Originally, constrained clauses of this flavor were introduced in [8] to reason about entailment in Herbrand models, and in particular in minimal models. The link between saturation and locality established in [6] and in the current paper opens a completely new avenue to explore the connection between locality and efficient reasoning in minimal models.

Acknowledgments. We thank the reviewers for their helpful comments. This work was partly supported by the German Research Council (DFG) as part of the Transre-

gional Collaborative Research Center “Automatic Verification and Analysis of Complex Systems” (SFB/TR 14 AVACS, see www.avacs.org).

References

- [1] L. Bachmair and H. Ganzinger. Rewrite-based equational theorem proving with selection and simplification. *J. of Logic and Computation*, 4(3):217–247, 1994.
- [2] D. Basin and H. Ganzinger. Complexity analysis based on ordered resolution. In *Proc. LICS’96*, pages 456–465. IEEE Computer Society Press, 1996.
- [3] D. Basin and H. Ganzinger. Automated complexity analysis based on ordered resolution. *Journal of the ACM*, 48(1):70–109, 2001.
- [4] H. Ganzinger. Relating semantic and proof-theoretic concepts for polynomial time decidability of uniform word problems. In *Proc. LICS’01*, pages 81–92. IEEE Computer Society Press, 2001.
- [5] R. Givan and D. McAllester. New results on local inference relations. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Third International Conference (KR’92)*, pages 403–412. Morgan Kaufmann Press, 1992.
- [6] M. Horbach and V. Sofronie-Stokkermans. Obtaining finite local theory axiomatizations via saturation. In P. Fontaine and R. Schmidt, editors, *Proc. FroCoS 2013*, LNCS 8152. Springer, 2013.
- [7] M. Horbach and C. Weidenbach. Deciding the inductive validity of $\forall\exists^*$ queries. In E. Grädel and R. Kahle, editors, *Proc. CSL 2009*, LNCS 5771, pages 332–347. Springer, 2009.
- [8] M. Horbach and C. Weidenbach. Superposition for Fixed Domains. *ACM Transactions on Computational Logic* 11(4), pages 27:1–27:35. 2010.
- [9] C. Ihlemann, S. Jacobs, and V. Sofronie-Stokkermans. On local reasoning in verification. In C. R. Ramakrishnan and J. Rehof, editors, *Proc. TACAS 2008*, LNCS 4963, pages 265–281. Springer, 2008.
- [10] C. Ihlemann and V. Sofronie-Stokkermans. On hierarchical reasoning in combinations of theories. In J. Giesl and R. Hähnle, editors, *Proc. IJCAR 2010*, LNAI 6173, pages 30–45. Springer, 2010.
- [11] T. Lev-Ami, N. Immerman, T.W. Reps, M. Sagiv, S. Srivastava, G. Yorsh. Simulating reachability using first-order logic with applications to verification of linked data structures. *Logical Methods in Computer Science* 5(2), 2009.
- [12] K. Klaessen. Expressing transitive closure for finite domains in pure first-order logic. Unpublished manuscript.
- [13] V. Sofronie-Stokkermans. Hierarchic reasoning in local theory extensions. In R. Nieuwenhuis, editor, *CADE-20*, LNAI 3632, pages 219–234. Springer, 2005.
- [14] V. Sofronie-Stokkermans. Locality results for certain extensions of theories with bridging functions. In R. A. Schmidt, editor, *Proc. CADE-22*, LNAI 5663, pages 67–83. Springer, 2009.
- [15] T. Wies, M. Muñoz, V. Kuncak. An efficient decision procedure for imperative tree data structures. In *CADE 2011*, LNCS 6803, pages 476–491, Springer 2011.
- [16] T. Wies, M. Muñoz, V. Kuncak. Deciding functional lists with sublist sets. In *Proc. VSTTE 2012*, LNCS 7152, pages 66–81, Springer 2012.