Modular Proof Systems for Partial Functions with Evans Equality

Harald Ganzinger †, Viorica Sofronie-Stokkermans, and Uwe Waldmann

Max-Planck-Institut für Informatik, Stuhlsatzenhausweg 85, 66123 Saarbrücken, Germany

Abstract

The paper presents a modular superposition calculus for the combination of first-order theories involving both total and partial functions. Modularity means that inferences are pure, only involving clauses over the alphabet of either one, but not both, of the theories. The calculus is shown to be complete provided that functions that are not in the intersection of the component signatures are declared as partial. This result also means that if the unsatisfiability of a goal modulo the combined theory does not depend on the totality of the functions in the extensions, the inconsistency will be effectively found. Moreover, we consider a constraint superposition calculus for the case of hierarchical theories and show that it has a related modularity property. Finally we identify cases where the partial models can always be made total so that modular superposition is also complete with respect to the standard (total function) semantics of the theories.

1 Introduction

This paper aims at providing new modularity results for refutational theorem proving in first-order logic with equality. In Nelson/Oppen-style combinations of two first-order theories $\mathcal{T}_1$ and $\mathcal{T}_2$ over signatures Σ_1 and Σ_2, inferences are *pure* in that all premises of an inference are clauses over only one of the signatures Σ_i where i depends on the inference. Therefore, no mixed formulae are ever generated when refuting goals represented by sets of pure formulae. What needs to be passed between the two theory modules are only universal

Email addresses: `sofronie@mpi-sb.mpg.de` (Viorica Sofronie-Stokkermans), `uwe@mpi-sb.mpg.de` (Uwe Waldmann).

Draft, submitted *8 August 2005*

formulae[1] over the intersection $\Sigma_1 \cap \Sigma_2$ of the two signatures. For stably infinite theories where, in addition, $\Sigma_1 \cap \Sigma_2$ consists of constants only, pure inference systems exist. This is one of the main consequences of Nelson and Oppen's results [23] (also see, e. g., Tinelli and Harandi [27] for additional clarification). The results have recently been extended to some situations when the theories $\mathcal{T}_1$ and $\mathcal{T}_2$ share also non-constant function symbols. Ghilardi [15] extended the completeness results for modular inference systems to a more general case of "compatibility" between the component theories $\mathcal{T}_i$. Future work might aim at weakening these compatibility requirements even further. In [26], Tinelli shows that similar modularity results are achieved if the theories share *all* their function symbols.

In this paper we take a different point of departure. We will consider arbitrary theory modules $\mathcal{T}_1$ and $\mathcal{T}_2$ and investigate what one loses in terms of completeness when superposition inferences are restricted to be pure. Superposition is refutationally complete for equational first-order logic, and by choosing term orderings appropriately (terms over $\Sigma_1 \cap \Sigma_2$ should be minimal in the term ordering), many, but not all, cases of impure inferences can be avoided. Impure inferences arise when one of the extensions $\Sigma_1 \setminus \Sigma_2$ or $\Sigma_2 \setminus \Sigma_1$ has additional non-constant function symbols. It is known that in such cases interpolants of implications of the form $\phi_1 \supset \phi_2$, with ϕ_i a Σ_i-formula, normally contain existential quantification. That means, that refutationally complete clausal theorem provers where existential quantifiers are skolemized need to pass clauses from $\mathcal{T}_1$ to $\mathcal{T}_2$ [from $\mathcal{T}_2$ to $\mathcal{T}_1$] containing function symbols not in Σ_2 [Σ_1]. In other words, inference systems are necessarily either incomplete or impure.

One of the main results of the paper is that if the extensions only introduce additional relations and partial functions,[2] a particular calculus of superposition for partial functions to be developed in this paper becomes a complete and modular proof system where inferences are pure. This result can be applied to problems where partial functions arise naturally. Alternatively we may think of this result as indicating what we lose if superposition is restricted to pure inferences. If a proof cannot be found in the pure system, a partial algebra model exists for the goal to be refuted. Conversely, if the inconsistency of a goal does not depend on the totality of the functions in the extensions, we will be able to find the inconsistency with the modular partial superposition calculus. There are interesting cases of problem classes where partial models can always be totalized and where the modular system is therefore in fact

[1] For Nelson/Oppen-style combination of theories, one even restricts the information exchange between theories to ground clauses over the intersection signature.

[2] A non-equational literal $p(t_1, \ldots, t_n)$ or $\neg\, p(t_1, \ldots, t_n)$, where p is a relation symbol, can be encoded as an equational literal $f_p(t_1, \ldots, t_n) \approx \mathit{true}_p$ or $\neg\, f_p(t_1, \ldots, t_n) \approx \mathit{true}_p$, where f_p is a function and true_p a total constant. Thus we will in the sequel not mention relations anymore.

complete (cf. Sect. 5).

1.1 Related work

The approach we present in this paper is based on two ideas: (i) consider extensions of a base theory with partial functions, (ii) show that in this case modular and hierarchical proof systems exist. We now explain how these ideas relate to previous work.

1.1.1 Evans Validity

We consider extensions of a base theory with partial functions. The semantics for partial functions we consider is known as "Evans validity". It was introduced, in the equational case, by Evans [10,11], while identifying situations when the uniform word problem in classes of algebras axiomatized by a set E of identities is decidable in PTIME.

We briefly present Evans' method and his motivation for giving this semantics for partial functions. Given a signature Σ, a presentation for Σ is a pair $\Pi = (G, R)$, where G is a set of generators and R is a set of relations (formulated in the signature Σ) between generators. The uniform word problem for a class of algebras axiomatized by a set E of identities is concerned with determining, for any presentation Π, which ground equations $u \approx v$ follow from E and R, i. e. when $E \cup R \models u \approx v$ holds. Evans' idea was to construct a "canonical" partial algebra P which satisfies E as well as all equations in R, and check if $u \approx v$ holds in P. For this, he started with the set $P(G, R)$ of all subterms occurring in $R \cup \{u \approx v\}$. This can be seen as a partial algebra, with operations defined in the natural way except for the fact that if $f \in \Sigma$ and $t_1, \dots, t_n \in P(G, R)$ then $f(t_1, \dots, t_n)$ is undefined in $P(G, R)$ if the term $f(t_1, \dots, t_n)$ is not in $P(G, R)$. Evans then identified subterms equal modulo $E \cup R$ using ground completion [3] of R together with certain ground instances of the theory clauses E dynamically derived from subterms in $P(G, R)$. The goal of the construction is to ensure that, in as much as the axioms in E are defined, they are satisfied in P. In addition, the functions in Σ must be defined in P in such a way that it is not possible, by the use of the axioms in E, to assign a value to some $f(p_1, \dots, p_m)$ which is not already defined in P. This last condition can be expressed as follows:

> If $s \approx f(s_1, \dots, s_n)$ is an axiom in E, and if for some substitution of elements in P the term s is defined in P and evaluates to p, and if $s_1, \dots, s_n$ are

[3] Before the concept was introduced by Knuth and Bendix in 1970.

defined in P and evaluate to $p_1, \ldots, p_n$, then $f(p_1, \ldots, p_n)$ must be defined in P and equal to p.

Thus, if a term t is undefined because some of its proper subterms are undefined, then t is "irrelevant" and can be excluded from further considerations. This reflects the way in which terms are replaced by equal terms in the ground completion process proposed by Evans. This link between rewriting, completion, and Evans equality was one of the reasons why in this paper we consider Evans equality for partial functions. What we propose is an extension of the completion algorithm to first-order clauses. Another reason is that embeddability conditions for partial algebras satisfying (in Evans' sense) sets of identities or Horn clauses were used [10,11,7] to obtain results on PTIME decidability of (uniform) word problems. We use similar embeddability results in Section 5 to establish a link between extensions with *partial* and extensions with *total* functions.

1.1.2 Modular reasoning in combinations of theories

The second main issue of this paper is modularity in automated theorem proving. This is a very important matter, as most of the reasoning problems which occur in computer science – especially in problems related to the verification of complex systems – can be reduced to reasoning in extensions and combinations of theories. One possibility is to integrate the knowledge about the individual components, taking into account the interaction between them. For this, "modularity" can be achieved by limiting interaction between the modules as much as possible, and using existing provers for the components as "black-boxes". In general interaction between modules cannot be ignored without losing completeness.

Let $\mathcal{T}_1, \mathcal{T}_2$ be two first-order theories in signatures Σ_1, Σ_2. Let Γ_1, Γ_2 be sets of clauses in the signatures Σ_1 resp. Σ_2. Assume that we want to show that $\mathcal{T}_1 \cup \Gamma_1 \cup \mathcal{T}_2 \cup \Gamma_2$ is satisfiable. In general it is not sufficient to check whether $\mathcal{T}_1 \cup \Gamma_1$ and $\mathcal{T}_2 \cup \Gamma_2$ are satisfiable: we need some information exchange between provers dealing with $\mathcal{T}_1 \cup \Gamma_1$ and $\mathcal{T}_2 \cup \Gamma_2$ respectively. By Craig's interpolation theorem for first-order logic we know that if $\mathcal{T}_1 \cup \Gamma_1 \cup \mathcal{T}_2 \cup \Gamma_2 \models \perp$ then there exists a formula ϕ containing only common symbols of $\mathcal{T}_1 \cup \Gamma_1$ and $\mathcal{T}_2 \cup \Gamma_2$ such that $\mathcal{T}_1 \cup \Gamma_1 \models \phi$ and $\mathcal{T}_2 \cup \Gamma_2 \cup \{\phi\} \models \perp$. However, ϕ can be an arbitrarily quantified first-order formula. It was proved that interpolants are always (ground) clauses if restrictions are imposed on the extensions to be taken into account, or on the shared theory:

- If the theories have disjoint signature, it can be proved that the interpolants are disjunctions of equalities between shared constants.
- In [26], Tinelli proved that if the theories $\mathcal{T}_1, \mathcal{T}_2$ share *all* function symbols

then the interpolants are always clauses (ground if Γ_1, Γ_2 are ground).
– Ghilardi [15] showed that a similar result holds if the theories are extensions of a shared theory and certain (model theoretic) compatibility conditions of these extensions with the shared theory are satisfied.

This is used in many methods for checking satisfiability of conjunctions of literals in combinations of theories. The Nelson-Oppen combination procedure [23] for instance, can be applied for combining decision procedures of *stably infinite* theories over disjoint signatures. As a preprocessing step, one purifies the problem by separating the theory symbols, thus obtaining a problem $\Gamma_1 \cup \Gamma_2$ consisting only of clauses with symbols in one, but not both, of the component theories. In a non-deterministic version of the procedure one then guesses (if possible) a combination of values for the shared variables which satisfies both Γ_1 and Γ_2. Arguments about stable infinity of the component theories are then used to infer that under these conditions the initial set of clauses is also satisfiable. Alternatively, in a "refutational" variant of the Nelson-Oppen procedure, one can analyze all inferences from the set $\Gamma_1 \cup \Gamma_2$ of purified clauses. This line of reasearch was pursued e. g. by Hillenbrand [18], who reestablished the correctness of the Nelson-Oppen combination procedure as a consequence of the superposition calculus [3]. Conditions when pure inferences are sufficient for checking unsatisfiability of purified goals in more general combinations of theories were identified by Tinelli and by Ghilardi [26,15].

The present paper changes the perspective compared with the approaches mentioned above. As in [26], we first consider extensions with relations and partial functions. However, in our paper the emphasis is on giving an efficient and modular superposition calculus for reasoning about partial functions. We then identify conditions under which the extension functions can be made total. Thus we do not need to use superposition for total functions, and are able to obtain complete modular resp. hierarchic calculi. Thus we relax some of the strong conditions imposed in [26] and [15] for obtaining similar results.

1.2 Structure of the Paper

In Sect. 2 we will describe the logic of partial functions we are working with. The logic is that of weak equality in the sense of Evans [11]. This logic allows one to specify undefinedness, but not definedness, of a function. (However we may specify a kind of relative definedness as explained below.) Then, in Sect. 3, we state and prove sound and refutationally complete a superposition calculus for clauses over signatures where functions can be declared as either total or partial. The calculus might be of independent interest for problem domains where partial functions arise in a natural manner. (That aspect, however, will not be explored any further in this paper as we are mainly interested in modularity.) We show that the calculus only admits pure inferences in cases of theory combinations where all functions that are not in the intersection of the signatures are declared as partial. In Sect. 4 we consider a variant of the calculus, called constraint partial superposition, suitable for hierarchical extensions $\mathcal{T}_1$ of a base theory $\mathcal{T}_0$. It differs from the previous calculus in that unification is replaced by generating equality constraints over the base theory. This system is modular in that no inferences involving base clauses (over Σ_0) need to be made. Rather, we may integrate any refutationally complete prover for $\mathcal{T}_0$ accepting the base clauses generated from non-base inferences and returning falsum whenever the accumulated set of base clauses is inconsistent with $\mathcal{T}_0$. In Sect. 5 we consider both shallow and local extensions of base theories, showing that for those classes of extensions constraint partial superposition is complete also with respect to the total algebra semantics of theories and goals. Finally Sect. 6 discusses related work.

This paper is an extended version of [14]. The considerations about the many-sorted case which were only mentioned in the short version of the paper are now fully presented.

2 Partial Functions with Evans Equality

Definition 1. A *many-sorted signature* $\Sigma = (S, \Omega^{\mathrm{T}}, \Omega^{\mathrm{P}})$ is a triple consisting of a non-empty set S of sorts, a set Ω^{T} of total function symbols, and a set Ω^{P} of partial function symbols.

Terms are built over Σ and a set V of variables. Each function symbol $f \in \Omega^{\mathrm{T}} \cup \Omega^{\mathrm{P}}$ comes with a unique declaration $f : \xi_1 \dots \xi_n \to \xi_0$ with $n \geq 0$ and $\xi_i \in S$; the sort ξ_0 is called the codomain of f.[4] Similarly, every variable $x \in V$ comes with a unique declaration $x : \xi$ for some $\xi \in S$.

Definition 2. The set $\mathrm{T}_\Sigma(V)_\xi$ of *terms of sort* ξ is inductively defined by $x \in \mathrm{T}_\Sigma(V)_\xi$ if $x : \xi \in V$ and $f(t_1, \dots, t_n) \in \mathrm{T}_\Sigma(V)_\xi$ if $f : \xi_1 \dots \xi_n \to \xi$ and $t_i \in \mathrm{T}_\Sigma(V)_{\xi_i}$; the union $\bigcup_{\xi \in S} \mathrm{T}_\Sigma(V)_\xi$ is denoted by $\mathrm{T}_\Sigma(V)$.

We assume that for every $\xi \in S$ the set $\mathrm{T}_\Sigma(\emptyset)_\xi$ contains at least one term consisting only of Ω^{T}-symbols. A *substitution* maps every variable $x \in V$ to a term with the same sort as x. An *equation* is a pair of terms, written as $s \approx t$, where s and t have the same sort.[5] We use $s \not\approx t$ as a shorthand for $\neg\, s \approx t$; in inference rules, the symbol $\dot{\approx}$ denotes either $\approx$ or $\not\approx$.

[4] If $S = \{\xi\}$ is a singleton, we use the shorthand notation f/n for an n-ary function symbol $f : \xi \dots \xi \to \xi$.

[5] For simplicity, we restrict to equality as the only predicate symbol. The extension to additional predicate symbols is obvious.

Definition 3. A *(partial) Σ-algebra* A consists of a non-empty set ξ_A for every $\xi \in S$, a total function $f_A : \xi_{1,A} \times \cdots \times \xi_{n,A} \to \xi_A$ for every $f : \xi_1 \ldots \xi_n \to \xi \in \Omega^{\mathrm{T}}$ and a partial function $g_A : \xi_{1,A} \times \cdots \times \xi_{n,A} \to \xi_A$ for every $g : \xi_1 \ldots \xi_n \to \xi \in \Omega^{\mathrm{P}}$.[6]

A Σ-algebra A is called *total* if g_A is a total function for every $g \in \Omega^{\mathrm{P}}$.

An *assignment* β into A is a function that maps every variable $x : \xi \in V$ to an element of ξ_A.

Definition 4. Given an algebra A and an assignment β into A, the *value* $(A, \beta)(t)$ *of a term* $t \in \mathrm{T}_\Sigma(V)_\xi$ is either an element of ξ_A or one of the two special values $\bot_{\mathrm{u}}$ (*"undefined"*) or $\bot_{\mathrm{i}}$ (*"irrelevant"*). It is defined as follows:

$(A, \beta)(x) = \beta(x)$
if x is a variable.

$(A, \beta)(f(t_1, \ldots, t_n)) = f_A(a_1, \ldots, a_n)$
if $(A, \beta)(t_i) = a_i \in \xi_{i,A}$ for all $i \in \{1, \ldots, n\}$
and $f_A(a_1, \ldots, a_n)$ is defined.

$(A, \beta)(f(t_1, \ldots, t_n)) = \bot_{\mathrm{u}}$
if $(A, \beta)(t_i) = a_i \in \xi_{i,A}$ for all $i \in \{1, \ldots, n\}$
and $f_A(a_1, \ldots, a_n)$ is undefined.

$(A, \beta)(f(t_1, \ldots, t_n)) = \bot_{\mathrm{i}}$
if $(A, \beta)(t_i) \in \{\bot_{\mathrm{u}}, \bot_{\mathrm{i}}\}$ for some $i \in \{1, \ldots, n\}$.

In other words, a term is irrelevant if one of its proper subterms is undefined.

To evaluate the truth of a formula, we use a three-valued logic with the values 1 (true), $\frac{1}{2}$ (undefined), and 0 (false). The truth values 1 and $\frac{1}{2}$ are called *positive*.

Definition 5. Given an algebra A and an assignment β into A, the *truth value of a formula* F w.r.t. A and β is denoted by $(A, \beta)(F)$. If F is an equation $s \approx t$, then $(A, \beta)(F) = 1$ if $(A, \beta)(s) = (A, \beta)(t) \in \xi_A$; $(A, \beta)(F) = \frac{1}{2}$ if $(A, \beta)(s) = (A, \beta)(t) = \bot_{\mathrm{u}}$ or $(A, \beta)(s) = \bot_{\mathrm{i}}$ or $(A, \beta)(t) = \bot_{\mathrm{i}}$; and otherwise $(A, \beta)(F) = 0$.

[6] We use $\xi_{i,A}$ as a shorthand for $(\xi_i)_A$.

For complex formulae, we have

$$
\begin{aligned}
&(A,\beta)(\bot) = 0,\\
&(A,\beta)(\top) = 1,\\
&(A,\beta)(F \wedge G) = \min\{(A,\beta)(F),(A,\beta)(G)\},\\
&(A,\beta)(F \vee G) = \max\{(A,\beta)(F),(A,\beta)(G)\},\\
&(A,\beta)(\neg F) = 1 - (A,\beta)(F),\\
&(A,\beta)(\forall x.F) = \min\{\, (A,\beta[x \mapsto a])(F) \mid x : \xi,\ a \in \xi_A \,\},\\
&(A,\beta)(\exists x.F) = \max\{\, (A,\beta[x \mapsto a])(F) \mid x : \xi,\ a \in \xi_A \,\}.
\end{aligned}
$$

Definition 6. An algebra A is a *(partial) model* of a formula F if $(A,\beta)(F) \geq \frac{1}{2}$ for every β, or in other words, if F is positive (i. e., true or undefined) w. r. t. A and β; it is a model of a set N of formulae if it is a model of every formula in N. A model is called *total* if it is a total algebra.

If A is a model of F, we say that F holds in A. A formula F follows from a set N of formulae (denoted by $N \models F$) if every model of N is a model of F. A set N of formulae is *satisfiable* if it has a model. Otherwise, it is called *unsatisfiable* or *inconsistent*; this is also denoted by $N \models \bot$.

Note that an algebra A is a model of a ground equation $s \approx t$ if both s and t are defined and equal in A, or if both are undefined, or if at least one of them is irrelevant; A is a model of $s \not\approx t$ unless both s and t are defined and different in A. It is easy to check that every ground clause C holds in an algebra A as soon as one term occurring in C is irrelevant in A. Intuitively, the ground instances of a clause that contain irrelevant terms are those instances that we choose to ignore.

Example 7. Let $\Omega^{\mathrm{T}} = \{\mathsf{nil}/0,\, \mathsf{cons}/2\}$, $\Omega^{\mathrm{P}} = \{\mathsf{car}/1,\, \mathsf{cdr}/1\}$, and let A be the algebra of finite lists with the usual interpretation of these symbols.

Then A is a model of $\forall x.\mathsf{cons}(\mathsf{car}(x), \mathsf{cdr}(x)) \approx x$: Suppose that $x : \xi$ is mapped to some $a \in \xi_A$. Then either both $\mathsf{car}_A(a)$ and $\mathsf{cdr}_A(a)$ are undefined, hence the value of $\mathsf{cons}(\mathsf{car}(x), \mathsf{cdr}(x))$ is irrelevant, and the equation has the truth value $\frac{1}{2}$. Or $\mathsf{car}_A(a)$ and $\mathsf{cdr}_A(a)$ are defined; in this case $\mathsf{cons}_A(\mathsf{car}_A(a), \mathsf{cdr}_A(a)) = a$, so the equation has the truth value 1. The truth value of the universally quantified formula is $\min\{\frac{1}{2}, 1\} = \frac{1}{2}$, therefore A is a model of the formula.

Since $\mathsf{car}_A(\mathsf{nil}_A)$ and $\mathsf{cdr}_A(\mathsf{nil}_A)$ are undefined, A is a model of both the formula $\mathsf{car}(\mathsf{nil}) \approx \mathsf{cdr}(\mathsf{nil})$ and its negation $\mathsf{car}(\mathsf{nil}) \not\approx \mathsf{cdr}(\mathsf{nil})$. It is not a model of $\mathsf{car}(\mathsf{nil}) \approx \mathsf{nil}$ (the left-hand side is undefined, the right-hand side is defined), it is, however, a model of $\mathsf{car}(\mathsf{car}(\mathsf{nil})) \approx \mathsf{nil}$ (the left-hand side is irrelevant).

Note that explicit [un-]definedness predicates are not present in this logic. To express that a term t is not defined, one can simply state that $t \not\approx t$. Expressing that t (not containing partial function symbols below the top) is defined is only possible if Σ contains appropriate total function symbols or can be extended by new symbols. For example, for an algebra B to be a model of $\forall x, y. \mathsf{car}(\mathsf{cons}(x, y)) \approx x$, car_B has to be defined for every b in the codomain of cons_B. Equations of this form implicitly express definedness requirements for partial functions.

Definition 8. A Σ-algebra is called *total-term-generated* if for every $a \in \xi_A$ there exists a ground term $t \in \mathrm{T}_\Sigma(\emptyset)_\xi$ consisting only of Ω^{T}-symbols such that $(A, \beta)(t) = a$. We write $N \models^{\mathrm{TG}} F$ if every total-term-generated model of N is a model of F.

Obviously, $N \models F$ implies $N \models^{\mathrm{TG}} F$. For refutational theorem proving, $\models$ and $\models^{\mathrm{TG}}$ are equivalent:

Proposition 9. *Let N be a set of universally quantified clauses. Then $N \models \bot$ if and only if $N \models^{\mathrm{TG}} \bot$.*

Proof. The "only if" part is trivial. For the "if" part assume that the Σ-algebra A is a model of N. Define a Σ-algebra B as follows: For $\xi \in S$ let ξ_B be the set of all elements $a \in \xi_A$ for which there is a ground term $t \in \mathrm{T}_\Sigma(\emptyset)_\xi$ consisting only of Ω^{T}-symbols such that $(A, \beta)(t) = a$. For $f : \xi_1 \ldots \xi_n \to \xi \in \Omega^{\mathrm{T}} \cup \Omega^{\mathrm{P}}$ and $b_i \in \xi_{i,B} \subseteq \xi_{i,A}$ let $f_B(b_1, \ldots, b_n) = f_A(b_1, \ldots, b_n)$ if $f_A(b_1, \ldots, b_n)$ is defined and contained in ξ_B; let $f_B(b_1, \ldots, b_n)$ be undefined otherwise. (Note that $f \in \Omega^{\mathrm{T}}$ and $b_i \in \xi_{i,B}$ implies that $f_A(b_1, \ldots, b_n)$ is defined and contained in ξ_B.)

It is now straightforward to verify that, for every assignment β into B and every literal $s \mathrel{\dot\approx} t$ occurring in a clause in N, $(A, \beta)(s \mathrel{\dot\approx} t) \geq \frac{1}{2}$ implies $(B, \beta)(s \mathrel{\dot\approx} t) \geq \frac{1}{2}$. Consequently, every clause that has positive truth value w. r. t. A must have positive truth value w. r. t. B. $\square$

Definition 10. We say that a substitution is *total* if no variable is mapped to a term containing a partial function symbol. If Q is a term or formula and σ is a total substitution, then $Q\sigma$ is a total instance of Q.

Definition 11. For a clause C, $\mathrm{tgi}(C)$ denotes the set of all total ground instances of C; for a set N of clauses, $\mathrm{tgi}(N) = \{\, C' \mid C \in N,\, C' \in \mathrm{tgi}(C) \,\}$.

Let A be a total-term-generated algebra and let V be a finite set of variables. Then for every assignment $\beta : V \to A$ there exists a total substitution $\sigma : V \to \mathrm{T}_\Sigma(\emptyset)$ such that $(A, \beta)(t) = (A, \gamma)(t\sigma)$ for all terms $t \in \mathrm{T}_\Sigma(V)$ and assignments $\gamma : V \to A$. Conversely, for every total substitution $\sigma : V \to \mathrm{T}_\Sigma(\emptyset)$ there exists an assignment $\beta : V \to A$ such that $(A, \beta)(t) = (A, \gamma)(t\sigma)$

for all terms $t \in \mathrm{T}_\Sigma(V)$ and assignments $\gamma : V \to A$. The following lemma is an immediate consequence of this fact:

Lemma 12. *Let N be a set of universally quantified clauses and let A be a total-term-generated algebra. Then A is a model of N if and only if A is a model of* $\mathrm{tgi}(N)$.

Convention 13. From now on, we will consider only the clausal fragment of this logic. As usual, all variables in a clause are implicitly universally quantified.

The theorem proving calculus described below will check whether a set N of clauses is inconsistent, that is, whether $N \models \bot$, where $\bot$ is the empty clause. The entailment problem "does a clause F follow from N" can be reduced to this refutation problem, but the reduction is a bit more complicated than in usual two-valued logic. The following example demonstrates the principal ideas of the reduction:

Example 14. Suppose that $\Omega^{\mathrm{T}} \supseteq \{\mathsf{c}/0, \mathsf{d}/0\}$ and $\Omega^{\mathrm{P}} \supseteq \{\mathsf{f}/1, \mathsf{g}/1\}$. We want to check whether $N \models \mathsf{f}(\mathsf{c}) \approx \mathsf{g}(\mathsf{d})$ for some set N of clauses. One might think that this is equivalent to $N \cup \{\mathsf{f}(\mathsf{c}) \not\approx \mathsf{g}(\mathsf{d})\} \models \bot$, but this is not true: If $N = \{\mathsf{f}(\mathsf{c}) \approx \mathsf{g}(\mathsf{d})\}$, then $N \models \mathsf{f}(\mathsf{c}) \approx \mathsf{g}(\mathsf{d})$, but still the set $N \cup \{\mathsf{f}(\mathsf{c}) \not\approx \mathsf{g}(\mathsf{d})\}$ has a model, namely one in which $\mathsf{f}(\mathsf{c})$ and $\mathsf{g}(\mathsf{d})$ are undefined. The statement $N \models \mathsf{f}(\mathsf{c}) \approx \mathsf{g}(\mathsf{d})$ holds if in each model of N either $\mathsf{f}(\mathsf{c})$ and $\mathsf{g}(\mathsf{d})$ are defined and equal, or both are undefined. Conversely, it does not hold if there is a model of N in which $\mathsf{f}(\mathsf{c})$ is defined and $\mathsf{g}(\mathsf{d})$ is undefined or defined and different from $\mathsf{f}(\mathsf{c})$, *or vice versa*. To translate the entailment problem into a *set of* refutation problems, we need therefore a new total function symbol $\mathsf{e}/0$: $N \models \mathsf{f}(\mathsf{c}) \approx \mathsf{g}(\mathsf{d})$ holds if and only if both $N \cup \{\mathsf{f}(\mathsf{c}) \approx \mathsf{e}, \mathsf{g}(\mathsf{d}) \not\approx \mathsf{e}\} \models \bot$ and $N \cup \{\mathsf{f}(\mathsf{c}) \not\approx \mathsf{e}, \mathsf{g}(\mathsf{d}) \approx \mathsf{e}\} \models \bot$.

3 Superposition for Partial Functions

The superposition calculus (Bachmair and Ganzinger [3]) is a saturation-based calculus for equational clauses that is refutationally complete and combines essentially the ideas of ordered resolution and unfailing Knuth-Bendix completion. The calculus is parameterized by a reduction ordering on terms (which is lifted to an ordering on literals and clauses). This ordering is used in two ways to reduce the search space of the calculus: Locally, inference rules are equipped with ordering restrictions so that inferences have to be performed only if they involve maximal terms[7] of maximal literals of clauses. Globally,

[7] Except for the *merging paramodulation* rule.

the ordering is used to define a redundancy criterion that allows us to delete or to simplify clauses.

In order to be sound for our logic of partial functions, the inference rules of the traditional superposition calculus must be modified in several ways. For instance, a literal $s \not\approx s$ may hold in an algebra – namely if s is undefined or irrelevant – so the *equality resolution* rule may be applied only if s is guaranteed to be defined. Similarly, replacement of equals by equals may be unsound: Assume that g is a partial function, $\mathsf{f}(\mathsf{g}(\mathsf{c}))$ is irrelevant in some algebra A, and d is defined, then $\mathsf{f}(\mathsf{g}(\mathsf{c})) \approx \mathsf{d}$ and $\mathsf{f}(\mathsf{g}(\mathsf{c})) \not\approx \mathsf{d}$ hold in A, but $\mathsf{d} \not\approx \mathsf{d}$ does not. Consequently, a term that is replaced using some inference rule may contain a partial function symbol at the top, but not below the top (so that it is guaranteed to be either defined or undefined, but not irrelevant). For the same reason, substitutions that introduce partial function symbols must be ruled out, so only total unifiers are permitted.

Inference System 15. The inference system of the *partial superposition calculus* consists of the inference rules *equality resolution*, *superposition*, *partial top-superposition*, *merging paramodulation*, and *factoring.*[8] Let us start the presentation of the inference rules with a few general conventions.

The partial superposition calculus is parameterized by a reduction ordering $\succ$ on terms that is total on ground terms and that has the property that every ground term over Ω^{T} is smaller than every ground term containing a symbol from Ω^{P} (for instance, a lexicographic path ordering where all symbols from Ω^{P} have higher precedence than symbols from Ω^{T}).[9]

To a positive literal $s \approx t$, we assign the multiset $\{s, t\}$, to a negative literal $\neg s \approx t$ the multiset $\{s, s, t, t\}$. The literal ordering $\succ_{\mathrm{L}}$ compares these multisets using the multiset extension of $\succ$. The clause ordering $\succ_{\mathrm{C}}$ compares clauses by comparing their multisets of literals using the multiset extension of $\succ_{\mathrm{L}}$.

A literal that is involved in an inference must be maximal in the respective clause (except for the literal $s_0 \approx s'_0$ in *merging paramodulation* and the literals $t_i \approx t'_i$ $(i > 1)$ in *partial top-superposition*). A positive literal that is involved in a *superposition*, *partial top-superposition*, or *merging paramodulation* inference must be strictly maximal in the respective clause (with the exceptions above). In inferences with two premises, the left premise is not greater than or equal to the right premise.

---

[8] The *merging paramodulation* rule could be replaced by the *equality factoring* rule [3]; the *factoring* rule is not subsumed by *equality factoring*, however, and would still be necessary for refutational completeness.

[9] Since we are interested in total ground instances only, this implies that a variable may be considered as smaller than every term containing a symbol from Ω^{P}.

Equality Resolution
$$\frac{C' \vee s \not\approx s'}{C'\sigma}$$

if s does not contain partial function symbols and σ is a total most general unifier of s and s'.

Superposition
$$\frac{D' \vee t \approx t' \qquad C' \vee s[u] \mathrel{\dot{\approx}} s'}{(D' \vee C' \vee s[t'] \mathrel{\dot{\approx}} s')\sigma}$$

if u is not a variable, t does not contain partial function symbols below the top, σ is a total most general unifier of t and u, $t\sigma \not\preceq t'\sigma$, $s\sigma \not\prec s'\sigma$, and, if $s \mathrel{\dot{\approx}} s'$ occurs positively or s is an Ω^{T}-term, then $s\sigma \not\preceq s'\sigma$.

Partial Top-Superposition
$$\frac{D' \vee t_1 \approx t'_1 \vee \ldots \vee t_n \approx t'_n \qquad C' \vee s \approx s'}{(D' \vee C' \vee s' \approx t'_1 \vee \ldots \vee s' \approx t'_n)\sigma}$$

if $n \geq 2$, s contains a partial function symbol at the top and no partial function symbols below the top, each t'_i contains a partial function symbol, σ is a total most general unifier of s and all t_i, $t_i\sigma \not\preceq t'_i\sigma$, $s\sigma \not\preceq s'\sigma$, and $s'\sigma \not\preceq t'_i\sigma$.[10]

Merging Paramodulation
$$\frac{D' \vee t \approx t' \qquad C' \vee s_0 \approx s'_0 \vee s \approx s'[u]}{(D' \vee C' \vee s_0 \approx s'_0 \vee s \approx s'[t'])\sigma}$$

if u is not a variable, t does not contain partial function symbols below the top, σ is a total most general simultaneous unifier of t and u and of s_0 and s, $t\sigma \not\preceq t'\sigma$, $s\sigma \not\preceq s'\sigma$, $s\sigma \not\preceq s'_0\sigma$, and $s'\sigma \not\preceq s'_0\sigma$,

Factoring
$$\frac{C' \vee s \approx s' \vee t \approx t'}{(C' \vee s \approx s')\sigma}$$

if σ is a total most general simultaneous unifier of s and t and of s' and t'.

Theorem 16. *The inference rules of the partial superposition calculus are sound w. r. t. $\models$ (and therefore also sound w. r. t. $\models^{\mathrm{TG}}$).*

Proof. We have to show that, whenever the premises of an inference hold in some algebra A, then the conclusion holds in A.

[10] *Partial top-superposition* corresponds to iterated *superposition* into the right premise, except that the intermediate conclusions may not be eliminated if they are redundant as defined below; in fact, it can be implemented that way. The *partial top-superposition* rule is needed in our proof of Lemma 26; the question whether the calculus is complete even without this rule is open, though.

Let us consider first the *equality resolution* rule. Suppose that A is a model of the clause $C = C' \vee s \not\approx s'$, where s is an Ω^{T}-term; let σ be a total unifier of s and s' and let β be an arbitrary assignment. Since σ is total, $x\sigma$ is an Ω^{T}-term and $(A,\beta)(x\sigma) \in \xi_A$ for every variable $x : \xi$. Define the assignment γ by $\gamma(x) = (A,\beta)(x\sigma)$. By assumption, $\frac{1}{2} \leq (A,\gamma)(C) = (A,\beta)(C\sigma) = (A,\beta)(C'\sigma \vee s\sigma \not\approx s'\sigma)$. Now note that $s\sigma = s'\sigma$ is an Ω^{T}-term, hence $(A,\beta)(s\sigma)$ and $(A,\beta)(s'\sigma)$ are defined and equal, therefore $(A,\beta)(s\sigma \not\approx s'\sigma) = 0$. Consequently, $(A,\beta)(C'\sigma) \geq \frac{1}{2}$. Since β could be chosen arbitrarily, A is a model of $C'\sigma$.

For the *superposition* rule assume that A is a model of the clauses $D = D' \vee t \approx t'$ and $C = C' \vee s[u] \mathrel{\dot\approx} s'$, where t does not contain Ω^{P}-symbols below the top. W.l.o.g., C and D have no common variables. Let σ be a total unifier of t and u and let β be an arbitrary assignment. Since σ is total, $x\sigma$ is an Ω^{T}-term and $(A,\beta)(x\sigma) \in \xi_A$ for every variable $x : \xi$. Define the assignment γ by $\gamma(x) = (A,\beta)(x\sigma)$. By assumption, $\frac{1}{2} \leq (A,\gamma)(C) = (A,\beta)(C\sigma) = (A,\beta)(C'\sigma \vee s\sigma[u\sigma] \mathrel{\dot\approx} s'\sigma)$ and $\frac{1}{2} \leq (A,\gamma)(D) = (A,\beta)(D\sigma) = (A,\beta)(D'\sigma \vee t\sigma \approx t'\sigma)$. If $(A,\beta)(C'\sigma) \geq \frac{1}{2}$ or $(A,\beta)(D'\sigma) \geq \frac{1}{2}$, it is obvious that the conclusion is positive w.r.t. A and β. Otherwise $(A,\beta)(s\sigma[u\sigma] \mathrel{\dot\approx} s'\sigma) \geq \frac{1}{2}$ and $(A,\beta)(t\sigma \approx t'\sigma) \geq \frac{1}{2}$. Since t does not contain Ω^{P}-symbols below the top, $(A,\beta)(t\sigma) \in \xi'_A \cup \{\bot_{\mathrm{u}}\}$. This leaves two possible reasons why $(A,\beta)(t\sigma \approx t'\sigma)$ is positive: If $(A,\beta)(t\sigma) = (A,\beta)(t'\sigma) \in \xi'_A \cup \{\bot_{\mathrm{u}}\}$, then clearly $(A,\beta)(s\sigma[t'\sigma] \mathrel{\dot\approx} s'\sigma) = (A,\beta)(s\sigma[t\sigma] \mathrel{\dot\approx} s'\sigma) = (A,\beta)(s\sigma[u\sigma] \mathrel{\dot\approx} s'\sigma) \geq \frac{1}{2}$. Otherwise $(A,\beta)(t'\sigma) = \bot_{\mathrm{i}}$, then $(A,\beta)(s\sigma[t'\sigma]) = \bot_{\mathrm{i}}$, hence $(A,\beta)(s\sigma[t'\sigma] \mathrel{\dot\approx} s'\sigma) = \frac{1}{2}$.

The soundness of the *partial top-superposition* and *merging paramodulation* rules is proved analogously.

Finally we consider the *factoring* rule. Let A be a model of the clause $C = C' \vee s \approx s' \vee t \approx t'$; let σ be a total simultaneous unifier of s and t and of s' and t', and let β be an arbitrary assignment. Define the assignment γ by $\gamma(x) = (A,\beta)(x\sigma)$. By assumption, $\frac{1}{2} \leq (A,\gamma)(C) = (A,\beta)(C\sigma) = (A,\beta)(C'\sigma \vee s\sigma \approx s'\sigma \vee t\sigma \approx t'\sigma)$. Clearly, $(A,\beta)(s\sigma \approx s'\sigma) = (A,\beta)(t\sigma \approx t'\sigma)$, hence $(A,\beta)(C'\sigma \vee s\sigma \approx s'\sigma) = (A,\beta)(C\sigma) \geq \frac{1}{2}$. Since β could be chosen arbitrarily, A is a model of the conclusion. $\square$

To keep the search space as small as possible, saturation-based inference systems are equipped with a global concept of redundancy that allows us to weaken the notion of saturation and to discard useless formulae. Let $\mathit{Red}^{\mathrm{C}}$ be a mapping from sets of formulae to sets of formulae and $\mathit{Red}^{\mathrm{I}}$ be a mapping from sets of formulae to sets of inferences. The sets $\mathit{Red}^{\mathrm{C}}(N)$ and $\mathit{Red}^{\mathrm{I}}(N)$ specify formulae and inferences considered unnecessary in the context of a given set N. Formulae in $\mathit{Red}^{\mathrm{C}}(N)$ may be removed from N during a theorem proving derivation, while inferences in $\mathit{Red}^{\mathrm{I}}(N)$ may be ignored. We emphasize that $\mathit{Red}^{\mathrm{C}}(N)$ need not be a subset of N and that $\mathit{Red}^{\mathrm{I}}(N)$ will usually also contain inferences whose premises are not in N.

Definition 17. The pair $\mathit{Red} = (\mathit{Red}^{\mathrm{I}}, \mathit{Red}^{\mathrm{C}})$ is called a *redundancy criterion* (with respect to an inference system *Inf* and a consequence relation $\models$) if the following conditions are satisfied for all sets of formulae N and N':

(i) $N \setminus \mathit{Red}^{\mathrm{C}}(N) \models \mathit{Red}^{\mathrm{C}}(N)$.
(ii) If $N \subseteq N'$, then $\mathit{Red}^{\mathrm{C}}(N) \subseteq \mathit{Red}^{\mathrm{C}}(N')$ and $\mathit{Red}^{\mathrm{I}}(N) \subseteq \mathit{Red}^{\mathrm{I}}(N')$.
(iii) If $N' \subseteq \mathit{Red}^{\mathrm{C}}(N)$, then $\mathit{Red}^{\mathrm{C}}(N) \subseteq \mathit{Red}^{\mathrm{C}}(N \setminus N')$ and $\mathit{Red}^{\mathrm{I}}(N) \subseteq \mathit{Red}^{\mathrm{I}}(N \setminus N')$.
(iv) If the conclusion of an *Inf*-inference ι is contained in N, then $\iota \in \mathit{Red}^{\mathrm{I}}(N)$.

Inferences in $\mathit{Red}^{\mathrm{I}}(N)$ and formulae in $\mathit{Red}^{\mathrm{C}}(N)$ are called redundant with respect to N.

Condition (i) requires that redundant formulae logically follow from the non-redundant ones. Conditions (ii) and (iii) indicate that redundant formulae and inferences must remain redundant if formulae are added or if redundant formulae are deleted. Finally, condition (iv) states that an inference is redundant with respect to N if its conclusion is already present in N (regardless of whether or not the premises are in N).

Definition 18. A ground clause C is called *redundant* w. r. t. a set N of ground clauses if it follows from a finite set of clauses in N that are smaller than C. A ground inference of the partial superposition calculus is called *redundant* w. r. t. a set N of ground clauses if one of its premises is redundant w. r. t. N or if its conclusion follows from a finite set of clauses in N that are smaller than the largest premise. The sets of redundant ground clauses and redundant ground inferences (of the partial superposition calculus) w. r. t. a set N of clauses are denoted by $\mathit{Red}^{\mathrm{C}}_{\mathrm{PSG}}(N)$ and $\mathit{Red}^{\mathrm{I}}_{\mathrm{PSG}}(N)$.

Lemma 19. *The pair* $(\mathit{Red}^{\mathrm{I}}_{\mathrm{PSG}}, \mathit{Red}^{\mathrm{C}}_{\mathrm{PSG}})$ *is a redundancy criterion for ground clauses and ground inferences of the partial superposition calculus w. r. t. the consequence relation* $\models$.

Proof. Condition (ii) of Def. 17 is obvious. Conditions (i) and (iii) follow from the well-foundedness of the reduction ordering $\succ$ (and König's Lemma). For condition (iv) observe that the conclusion of every ground inference of the partial superposition calculus is smaller than its largest premise. $\square$

For general clauses and inferences, redundancy is defined by lifting:

Definition 20. Let ι be an inference with premises $C_1, \dots, C_n$ and conclusion C; let ι' be an inference with ground premises $C'_1, \dots, C'_n$ and conclusion C'. We say that ι' is a *total ground instance* of ι if σ is a total substitution,

$C_i\sigma = C_i'$, and $C\sigma = C'$. The set of all total ground instances of ι is denoted by $\mathrm{tgi}(\iota)$.

Definition 21. A clause C is *redundant* w. r. t. a set N of clauses if $\mathrm{tgi}(C) \subseteq \mathit{Red}^{\mathrm{C}}_{\mathrm{PSG}}(\mathrm{tgi}(N))$; in inference ι is *redundant* w. r. t. a set N of clauses if $\mathrm{tgi}(\iota) \subseteq \mathit{Red}^{\mathrm{I}}_{\mathrm{PSG}}(\mathrm{tgi}(N))$. The sets of redundant clauses and redundant inferences (of the partial superposition calculus) w. r. t. a set N of clauses are denoted by $\mathit{Red}^{\mathrm{C}}_{\mathrm{PS}}(N)$ and $\mathit{Red}^{\mathrm{I}}_{\mathrm{PS}}(N)$.

As $M \subseteq M'$ implies $\mathit{Red}^{\mathrm{I}}_{\mathrm{PSG}}(M) \subseteq \mathit{Red}^{\mathrm{I}}_{\mathrm{PSG}}(M')$, we obtain $\mathit{Red}^{\mathrm{C}}_{\mathrm{PSG}}(\mathrm{tgi}(N) \setminus \mathrm{tgi}(N')) \subseteq \mathit{Red}^{\mathrm{C}}_{\mathrm{PSG}}(\mathrm{tgi}(N \setminus N'))$. Furthermore, it is fairly easy to see that $\mathrm{tgi}(N) \setminus \mathit{Red}^{\mathrm{C}}_{\mathrm{PSG}}(\mathrm{tgi}(N)) \subseteq \mathrm{tgi}(N \setminus \mathit{Red}^{\mathrm{C}}_{\mathrm{PS}}(N))$. Using these two results we can prove the following lemma:

Lemma 22. *The pair* $(\mathit{Red}^{\mathrm{I}}_{\mathrm{PS}}, \mathit{Red}^{\mathrm{C}}_{\mathrm{PS}})$ *is a redundancy criterion with respect to the inference system of the partial superposition calculus and the consequence relation* $\models^{\mathrm{TG}}$.

Proof. For condition (i) of Def. 17 we have to show that $N \setminus \mathit{Red}^{\mathrm{C}}_{\mathrm{PS}}(N) \models^{\mathrm{TG}} \mathit{Red}^{\mathrm{C}}_{\mathrm{PS}}(N)$. By Def. 8 and Lemma 12, it is sufficient to show that $\mathrm{tgi}(N \setminus \mathit{Red}^{\mathrm{C}}_{\mathrm{PS}}(N)) \models \mathrm{tgi}(\mathit{Red}^{\mathrm{C}}_{\mathrm{PS}}(N))$. Let D be an arbitrary ground clause from $\mathrm{tgi}(\mathit{Red}^{\mathrm{C}}_{\mathrm{PS}}(N))$. Since $D \in \mathit{Red}^{\mathrm{C}}_{\mathrm{PSG}}(\mathrm{tgi}(N))$, we have $\mathrm{tgi}(N) \setminus \mathit{Red}^{\mathrm{C}}_{\mathrm{PSG}}(\mathrm{tgi}(N)) \models D$ and consequently $\mathrm{tgi}(N \setminus \mathit{Red}^{\mathrm{C}}_{\mathrm{PS}}(N)) \models D$. Therefore $N \setminus \mathit{Red}^{\mathrm{C}}_{\mathrm{PS}}(N) \models^{\mathrm{TG}} \mathit{Red}^{\mathrm{C}}_{\mathrm{PS}}(N)$.

The proof of property (ii) is trivial. For condition (iii) note that $N' \subseteq \mathit{Red}^{\mathrm{C}}_{\mathrm{PS}}(N)$ implies $\mathrm{tgi}(N') \subseteq \mathit{Red}^{\mathrm{C}}_{\mathrm{PSG}}(\mathrm{tgi}(N))$. Now let $D \in \mathit{Red}^{\mathrm{C}}_{\mathrm{PS}}(N)$, then $\mathrm{tgi}(D) \subseteq \mathit{Red}^{\mathrm{C}}_{\mathrm{PSG}}(\mathrm{tgi}(N)) \subseteq \mathit{Red}^{\mathrm{C}}_{\mathrm{PSG}}(\mathrm{tgi}(N) \setminus \mathrm{tgi}(N')) \subseteq \mathit{Red}^{\mathrm{C}}_{\mathrm{PSG}}(\mathrm{tgi}(N \setminus N'))$ and thus $\mathit{Red}^{\mathrm{C}}_{\mathrm{PS}}(N) \subseteq \mathit{Red}^{\mathrm{C}}_{\mathrm{PS}}(N \setminus N')$. The proof for inferences works analogously.

For condition (iv) let ι be an inference whose conclusion is contained in N. Then the conclusions of all inferences in $\mathrm{tgi}(\iota)$ are contained in $\mathrm{tgi}(N)$. As $\mathrm{tgi}(\iota) \subseteq \mathit{Red}^{\mathrm{I}}_{\mathrm{PSG}}(\mathrm{tgi}(N))$, the inference ι is contained in $\mathit{Red}^{\mathrm{I}}_{\mathrm{PS}}(N)$. This proves condition (iv). $\square$

Definition 23. A set N of clauses is called *saturated up to redundancy* if all inferences between clauses in N are redundant w. r. t. N.

A saturated set can be obtained as the limit of a fair theorem proving derivation (see Bachmair, Ganzinger, and Waldmann [5] for the details).

We will show that the partial superposition calculus is refutationally complete, that is, that a saturated set of clauses has a model if and only if it does not contain the empty clause. The "only if" part of this proposition is of course trivial. For the "if" part, we have to construct a model of a saturated set N. This model is represented by a convergent term rewrite system or, equivalently, by an equational theory. For every sort $\xi \in S$, the set ξ_A consists of all ground normal forms of the rewrite system that are Ω^{T}-terms of sort ξ (or, equivalently, of the congruence classes of all ground Ω^{T}-terms of sort ξ). Given such a model, a ground term is defined if its normal form is an Ω^{T}-term; it is undefined if all its immediate subterms have normal forms that are Ω^{T}-terms, but the term itself does not; it is irrelevant if some of its subterms do not have normal forms that are Ω^{T}-terms.

The rewrite system is constructed from the set $\overline{N}$ of total ground instances of clauses in N. Starting with an empty interpretation all such instances are inspected in ascending order w. r. t. the clause ordering. If a clause is false and irreducible in the interpretation constructed so far and if it has a strictly maximal literal $s \approx s'$ with $s \succ s'$, then $s \approx s'$ is turned into a rewrite rule and added to the interpretation (Bachmair and Ganzinger [3]).

Let N be a set of clauses not containing $\bot$. Using induction on the clause ordering we define sets of rewrite rules E_C and R_C for all $C \in \overline{N}$ as follows:

Assume that E_D has already been defined for all $D \in \overline{N}$ with $D \prec_{\mathrm{C}} C$. Then $R_C = \bigcup_{D \prec_{\mathrm{C}} C} E_D$. The set E_C contains the rewrite rule $s \to s'$ if

(a) $C = C' \vee s \approx s'$.
(b) $s \approx s'$ is strictly maximal in C.
(c) $s \succ s'$.
(d) C is false in R_C.
(e) C' is false in $R_C \cup \{s \to s'\}$.
(f) s is irreducible w. r. t. R_C and contains no Ω^{P}-symbols below the top.
(g) the R_C-normal form of s' contains no Ω^{P}-symbols.
(h) no clause $D \in \overline{N}$ with $D \prec_{\mathrm{C}} C$ is false in $R_C \cup \{s \to s'\}$.

In this case, C is called productive. Otherwise $E_C = \emptyset$. Finally, $R_\infty = \bigcup_{D \in \overline{N}} E_D$.

The sequence of interpretations generated in this way has two monotonicity properties:

Lemma 24. *If a clause C has positive truth value in R_C, then it has positive truth value in R_∞ and R_D for every $D \succ_{\mathrm{C}} C$.*

Proof. By condition (h) of the model construction. □

Lemma 25. *If a clause $C = C' \vee s \approx s'$ is productive then C is true and C' is false in R_∞ and R_D for every $D \succ_{\mathrm{C}} C$.*

Proof. By the ordering conditions, no rule in $R_\infty \setminus (R_C \cup E_C)$ can be used to rewrite a term in C'. □

It is clear from these two monotonicity properties that every clause in $\overline{N}$ has positive truth value in the limit interpretation R_∞ if either it has positive truth value at the time where it is inspected or if it is productive. It remains to show that every ground instance in $\overline{N}$ falls into one of these two classes if N is saturated up to redundancy and does not contain the empty clause.

Lemma 26. *Let N be a set of clauses that is saturated up to redundancy and does not contain the empty clause. Then we have for every total ground instance $C\theta \in \overline{N}$:*

(i) $E_{C\theta} = \emptyset$ if and only if $C\theta$ has positive truth value in $R_{C\theta}$.
(ii) $C\theta$ has positive truth value in R_∞ and in R_D for every $D \succ_{\mathrm{C}} C\theta$.
(iii) If $C\theta$ is redundant w. r. t. $\overline{N}$, then $E_{C\theta} = \emptyset$.

Proof. We prove the three properties (i)–(iii) simultaneously by well-founded induction on the clause ordering $\succ_{\mathrm{C}}$. Let $C\theta$ be a total ground instance in $\overline{N}$. By the induction hypothesis, we assume that (i)–(iii) are satisfied for all clauses in $\overline{N}$ that are smaller than $C\theta$. Note that the "if" part of (i) is obvious from the model construction and that condition (ii) follows from (i) by Lemma 24 and Lemma 25. So it remains to show that $C\theta$ satisfies (iii) and the "only if" part of (i). To this end, we test first whether $C\theta$ is redundant w. r. t. $\overline{N}$ or whether $x\theta$ is reducible by $R_{C\theta}$ for some variable x in C. The remainder of the proof is a case analysis over the syntactical structure of $C\theta$ (with most cases corresponding to inference rules of the partial superposition calculus).

Case 1: $C\theta$ is redundant w. r. t. $\overline{N}$.

If $C\theta$ is redundant w. r. t. $\overline{N}$, then it follows from clauses in $\overline{N}$ that are smaller than $C\theta$. By part (ii) of the the induction hypothesis, these clauses have positive truth value in $R_{C\theta}$. So $C\theta$ has positive truth value in $R_{C\theta}$, consequently $E_{C\theta} = \emptyset$, and (i)–(iii) are satisfied.

In the remaining cases, it suffices to show that $C\theta$ satisfies the "only if" part of (i).

Case 2: $x\theta$ is reducible by $R_{C\theta}$.

Suppose that $C\theta$ does not fall into Case 1 and that there is a variable x occurring in C such that $x\theta$ is reducible by $R_{C\theta}$, say $x\theta \rightarrow_{R_{C\theta}} w$. Let the total substitution θ' be defined by $x\theta' = w$ and $y\theta' = y\theta$ for every variable $y \neq x$. The clause $C\theta'$ is smaller than $C\theta$. By part (ii) of the induction hypothesis, it has positive truth value in $R_{C\theta}$. As every literal of $C\theta$ has the same truth value $R_{C\theta}$ as the corresponding literal of $C\theta'$, $C\theta$ has positive truth value in $R_{C\theta}$.

Case 3: $C\theta$ contains a maximal negative literal.

Suppose that $C\theta$ does not fall into Cases 1 or 2 and that $C\theta = C'\theta \vee s\theta \not\approx s'\theta$, where $s\theta \not\approx s'\theta$ is maximal in $C\theta$. If $s\theta \approx s'\theta$ is false or undefined in $R_{C\theta}$, then $C\theta$ is true or undefined in $R_{C\theta}$ and we are done. So assume that $s\theta \approx s'\theta$ is true in $R_{C\theta}$, that is, $s\theta$ and $s'\theta$ have the same Ω^{T}-term as $R_{C\theta}$-normal form. Without loss of generality, $s\theta \succeq s'\theta$.

Case 3.1: $s\theta = s'\theta$ and s is an Ω^{T}-term.

If $s\theta = s'\theta$ and s is an Ω^{T}-term, then there is an *equality resolution* inference

$$\frac{C'\theta \vee s\theta \not\approx s'\theta}{C'\theta}\ .$$

This is an instance of an *equality resolution* inference from C. By saturation up to redundancy, it is redundant, hence its conclusion follows from clauses in $\overline{N}$ that are smaller than $C\theta$. By the induction hypothesis, these clauses have positive truth value in $R_{C\theta}$. Thus $C'\theta$ and $C\theta$ have positive truth value in $R_{C\theta}$.

Case 3.2: $s\theta \succ s'\theta$ or s contains an Ω^{P}-symbol.

If $s\theta$ and $s'\theta$ can be rewritten to the same Ω^{T}-term u, and $s\theta \succ s'\theta$ or s contains an Ω^{P}-symbol then $s\theta$ must be reducible by some rule in some $E_{D\theta} \subseteq R_{C\theta}$. (Without loss of generality we assume that C and D are variable disjoint; so we can use the same substitution θ.) Let $D\theta = D'\theta \vee t\theta \approx t'\theta$ with $E_{D\theta} = \{t\theta \to t'\theta\}$. By part (iii) of the induction hypothesis, $D\theta$ is not redundant, and by Lemma 25, $D'\theta$ is false in $R_{C\theta}$.

Note that $t\theta$ cannot occur in $s\theta$ at or below a variable position of s, say $x\theta = w[t\theta]$, since otherwise $C\theta$ would be subject to Case 2 above. Consequently, the *superposition* inference

$$\frac{D'\theta \vee t\theta \approx t'\theta \qquad C'\theta \vee s\theta[t\theta] \not\approx s'\theta}{D'\theta \vee C'\theta \vee s\theta[t'\theta] \not\approx s'\theta}$$

is a ground instance of a *superposition* inference from D and C. By saturation up to redundancy, its conclusion follows from clauses in $\overline{N}$ that are smaller than $C\theta$. By the induction hypothesis, these clauses have positive truth value in $R_{C\theta}$, thus $D'\theta \vee C'\theta \vee s\theta[t'\theta] \not\approx s'\theta$ has positive truth value in $R_{C\theta}$. Since $D'\theta$ and $s\theta[t'\theta] \not\approx s'\theta$ are false in $R_{C\theta}$, both $C'\theta$ and $C\theta$ must have positive truth value.

Case 4: $C\theta$ does not contain a maximal negative literal.

Suppose that $C\theta$ does not fall into Cases 1 to 3. Then $C\theta$ can be written as

$C'\theta \vee s\theta \approx s'\theta$, where $s\theta \approx s'\theta$ is a maximal literal of $C\theta$. If $E_{C\theta} = \{s\theta \to s'\theta\}$ or $C'\theta$ has positive truth value in $R_{C\theta}$ or $s\theta = s'\theta$, then there is nothing to show, so assume that $E_{C\theta} = \emptyset$ and that $C'\theta$ is false in $R_{C\theta}$. Without loss of generality, $s\theta \succ s'\theta$.

Case 4.1: $s\theta \approx s'\theta$ is maximal in $C\theta$, but not strictly maximal.

If $s\theta \approx s'\theta$ is maximal in $C\theta$, but not strictly maximal, then $C\theta$ can be written as $C''\theta \vee t\theta \approx t'\theta \vee s\theta \approx s'\theta$, where $t\theta = s\theta$ and $t'\theta = s'\theta$. In this case, there is a *factoring* inference

$$\frac{C''\theta \vee t\theta \approx t'\theta \vee s\theta \approx s'\theta}{C''\theta \vee s\theta \approx s'\theta}$$

This inference is a ground instance of an inference from C. By saturation, its conclusion has positive truth value in $R_{C\theta}$, so $C\theta$ must also have positive truth value in $R_{C\theta}$.

Case 4.2: $s\theta \approx s'\theta$ is strictly maximal in $C\theta$ and $s\theta$ is reducible.

Suppose that $s\theta \approx s'\theta$ is strictly maximal in $C\theta$ and $s\theta$ is reducible by some rule in $E_{D\theta} \subseteq R_{C\theta}$. Let $D\theta = D'\theta \vee t\theta \approx t'\theta$ and $E_{D\theta} = \{t\theta \to t'\theta\}$. Since $D\theta$ is productive, it is not redundant and $D'\theta$ is false in $R_{C\theta}$. We can now proceed in essentially the same way as in Case 3.2: If $t\theta$ occurred in $s\theta$ at or below a variable position of s, say $x\theta = w[t\theta]$, then $C\theta$ would be subject to Case 2 above. Otherwise, the *superposition* inference

$$\frac{D'\theta \vee t\theta \approx t'\theta \qquad C'\theta \vee s\theta[t\theta] \approx s'\theta}{D'\theta \vee C'\theta \vee s\theta[t'\theta] \approx s'\theta}$$

is a ground instance of a *superposition* inference from D and C. By saturation up to redundancy, its conclusion has positive truth value in $R_{C\theta}$. Since $D'\theta$ and $C'\theta$ are false in $R_{C\theta}$, $s\theta[t'\theta] \approx s'\theta$ must have positive truth value in $R_{C\theta}$. On the other hand, $t\theta \approx t'\theta$ is true in $R_{C\theta}$, so $s\theta[t\theta] \approx s'\theta$ and hence $C\theta$ have positive truth value in $R_{C\theta}$.

Case 4.3: $s\theta$ contains an Ω^{P}-symbol below the top.

Suppose that $s\theta$ contains an Ω^{P}-symbol below the top. If the subterm at that position is reducible, then $C\theta$ is subject to Case 4.2 above. Otherwise $s\theta$ is irrelevant, hence $s\theta \approx s'\theta$ and $C\theta$ are undefined in $R_{C\theta}$.

Case 4.4: The $R_{C\theta}$-normal form of $s'\theta$ contains an Ω^{P}-symbol.

Assume that the $R_{C\theta}$-normal form of $s'\theta$ contains an Ω^{P}-symbol. Then $s\theta$ and $s'\theta$ must also contain Ω^{P}-symbols. If $s\theta$ is reducible, then $C\theta$ is subject to

Case 4.2 above. Otherwise, both $s\theta$ and $s'\theta$ are undefined or irrelevant in $R_{C\theta}$, hence $s\theta \approx s'\theta$ and $C\theta$ are undefined in $R_{C\theta}$.

Case 4.5: Otherwise.

Suppose that $s\theta \approx s'\theta$ is strictly maximal in $C\theta$, $s\theta$ is irreducible by $R_{C\theta}$ and contains no Ω^{P}-symbols below the top, and the $R_{C\theta}$-normal form of $s'\theta$ contains no Ω^{P}-symbols. If $E_{C\theta} = \{s\theta \to s'\theta\}$ or if $C\theta$ has positive truth value in $R_{C\theta}$, there is nothing to show. So there are only two possibilities left: Condition (e) or condition (h) of the model contruction must be violated. In other words, $C'\theta$ has positive truth value in $R_{C\theta} \cup \{s\theta \to s'\theta\}$, or some $D\theta \prec_{\mathrm{C}} C\theta$ in $\overline{N}$ is false in $R_{C\theta} \cup \{s\theta \to s'\theta\}$.

Case 4.5.1: $C'\theta$ has positive truth value in $R_{C\theta} \cup \{s\theta \to s'\theta\}$.

Let us assume that $C\theta$ is false in $R_{C\theta}$ and $C'\theta$ is true or undefined in $R_{C\theta} \cup \{s\theta \to s'\theta\}$. It is impossible that the truth value of a positive literal in $C'\theta$ changes from false to undefined by adding the rewrite rule $s\theta \to s'\theta$, and it is also impossible that the truth value of a negative literal in $C'\theta$ changes from false to true or undefined. We can conclude that $C'\theta = C''\theta \vee s_0\theta \approx s_0'\theta$, where the literal $s_0\theta \approx s_0'\theta$ is true in $R_{C\theta} \cup \{s\theta \to s'\theta\}$ and false in $R_{C\theta}$. In other words, $s_0\theta \downarrow_{R_{C\theta}\cup\{s\theta\to s'\theta\}} s_0'\theta$, but not $s_0\theta \downarrow_{R_{C\theta}} s_0'\theta$. Consequently, there is a rewrite proof of $s_0\theta \to^* u \leftarrow^* s_0'\theta$ by $R_{C\theta} \cup \{s\theta \to s'\theta\}$ in which the rule $s\theta \to s'\theta$ is used at least once. Without loss of generality we assume that $s_0\theta \succeq s_0'\theta$. Since $s\theta \approx s'\theta \succ_{\mathrm{L}} s_0\theta \approx s_0'\theta$ and $s\theta \succ s'\theta$ we can conclude that $s\theta \succeq s_0\theta \succ s_0'\theta$. But then there is only one possibility how the rule $s\theta \to s'\theta$ can be used in the rewrite proof: We must have $s\theta = s_0\theta$ and the rewrite proof must have the form $s_0\theta \to s'\theta \to^+ u \leftarrow^* s_0'\theta$, where the first step uses $s\theta \to s'\theta$ and all other steps use rules from $R_{C\theta}$. Consequently, $s'\theta$ is reducible by some rule in $E_{D\theta} \subseteq R_{C\theta}$. Let $D\theta = D'\theta \vee t\theta \approx t'\theta$ and $E_{D\theta} = \{t\theta \to t'\theta\}$. We can now proceed in essentially the same way as in Case 3.2: If $t\theta$ occurred in $s\theta$ at or below a variable position of s, say $x\theta = w[t\theta]$, then $C\theta$ would be subject to Case 2 above. Otherwise, the *merging paramodulation* inference

$$\frac{D'\theta \vee t\theta \approx t'\theta \qquad C''\theta \vee s_0\theta \approx s_0'\theta \vee s\theta \approx s'\theta[u\theta]}{D'\theta \vee C''\theta \vee s_0\theta \approx s_0'\theta \vee s\theta \approx s'\theta[t'\theta]}$$

is a ground instance of a *merging paramodulation* inference from D and C. By saturation up to redundancy, its conclusion has positive truth value in $R_{C\theta}$. Since $D'\theta$ and $C'\theta$ are false in $R_{C\theta}$, $s\theta \approx s'\theta[t'\theta]$ must have positive truth value in $R_{C\theta}$. On the other hand, $t\theta \approx t'\theta$ is true in $R_{C\theta}$, so $s\theta \approx s'\theta[u\theta]$ and hence $C\theta$ have positive truth value in $R_{C\theta}$, contradicting our assumption.

Case 4.5.2: Some $D\theta \prec_{\mathrm{C}} C\theta$ is false in $R_{C\theta} \cup \{s\theta \to s'\theta\}$.

If there are clauses in $\overline{N}$ that are smaller than $C\theta$ and false in $R_{C\theta} \cup \{s\theta \to$

$s'\theta\}$, let $D\theta$ be the smallest such clause. By the induction hypothesis, $D\theta$ has positive truth value in $R_{C\theta}$. If $D\theta$ becomes false by adding $s\theta \to s'\theta$ to $R_{C\theta}$, $D\theta$ must contain at least one literal whose left-hand side equals $s\theta$ and whose right-hand side is undefined in $R_{C\theta}$ (and contains therefore Ω^{P}-symbols); moreover no term in $D\theta$ can be irrelevant in $R_{C\theta}$. Let $t_1\theta \approx t'_1\theta$ be a maximal literal of $D\theta$. We observe three things: First, $t_1\theta$ must equal $s\theta$. Second, $t_1\theta \approx t'_1\theta$ must be strictly maximal. Otherwise there is a *factoring* inference from $D\theta$, and by redundancy of this inference $D\theta$ cannot be the smallest clause that becomes false by adding $s\theta \to s'\theta$ to $R_{C\theta}$. Third, $t'_1\theta$ must be undefined in $R_{C\theta}$. Otherwise, there would be another literal $t_0\theta \approx t'_0\theta$ in $D\theta$ with $t_0\theta = s\theta$ and $t'_0\theta$ undefined in $R_{C\theta}$, and since $t'_1\theta$ would be reducible by $R_{C\theta}$, there would be a *merging paramodulation* inference with $D\theta$ as the second premise, whose redundancy contradicts again the minimality of $D\theta$.

Let $D\theta = D'\theta \vee t_1\theta \approx t'_1\theta \vee \ldots \vee t_n\theta \approx t'_n\theta$, where all $t_i\theta$ equal $s\theta$ and where all $t'_i\theta$ are undefined in $R_{C\theta}$. The *superposition* inference (if $n = 1$) or *partial top-superposition* inference (if $n \geq 2$)

$$\frac{D'\theta \vee t_1\theta \approx t'_1\theta \vee \ldots \vee t_n\theta \approx t'_n\theta \qquad C'\theta \vee s\theta \approx s'\theta}{D'\theta \vee C'\theta \vee s'\theta \approx t'_1\theta \vee \ldots \vee s'\theta \approx t'_n\theta}$$

is a ground instance of a *superposition* or *partial top-superposition* inference from D and C. By saturation up to redundancy, its conclusion has positive truth value in $R_{C\theta}$. Since $D'\theta$ and $C'\theta$ are false in $R_{C\theta}$, one of the literals $s'\theta \approx t'_i\theta$ must have positive truth value in $R_{C\theta}$. Since $s'\theta$ is defined, however, this implies that $t'_i\theta$ is defined, contradicting our assumption. This concludes the proof of the lemma. □

Theorem 27. *The partial superposition calculus is refutationally complete.*

Proof. We have to show that a saturated set N of clauses has a model if and only if does not contain the empty clause.

If N contains the empty clause, then obviously it does not have a model. Otherwise, the rewrite system R_∞ constructed above gives us a Σ-algebra A: For $\xi \in S$, the set ξ_A consists of all ground normal forms of R_∞ that are Ω^{T}-terms of sort ξ (or, equivalently, of the congruence classes of all ground Ω^{T}-terms of sort ξ). A function $f_A : \xi_{1,A} \times \cdots \times \xi_{n,A} \to \xi_A$ maps the terms $t_1, \ldots, t_n$ to the R_∞-normal form of $f(t_1, \ldots, t_n)$ if this is an Ω^{T}-term, it is undefined otherwise. By part (ii) of Lemma 26, A is a model of all total ground instances of clauses in N, hence by Lemma 12, it is a model of N. □

There are alternative ways of dealing with partial functions in automated theorem proving, notably by encoding a partial function f/n as an $(n+1)$-ary relation r together with a clause $\neg r(x_1, \ldots, x_n, y) \vee \neg r(x_1, \ldots, x_n, y') \vee y \approx y'$. One may ask whether partial superposition has any advantages over such an encoding. First, it is clear that the flattening of terms resulting from the relational encoding will generally make it more difficult to detect simplification opportunities. Second, the strengthened ordering restrictions of partial superposition reduce the number of possible inferences. The following trivial example illustrates this:

Example 28. Let $\Omega^{\mathrm{T}} = \{\mathsf{c}/0,\, \mathsf{d}/0,\, \mathsf{e}/0\}$, let $\Omega^{\mathrm{P}} = \{\mathsf{f}/1\}$, and suppose that N contains the clauses

$$\begin{aligned} \mathsf{f}(\mathsf{c}) &\approx \mathsf{d} \\ \mathsf{f}(\mathsf{c}) &\approx \mathsf{e} \\ \mathsf{d} &\not\approx \mathsf{e} \end{aligned}$$

where $\mathsf{c} \succ \mathsf{d} \succ \mathsf{e}$. Partial superposition derives $\mathsf{d} \approx \mathsf{e}$ from the first two clauses, then $\mathsf{e} \not\approx \mathsf{e}$, and then the empty clause. This whole process is completely deterministic: no other inferences are possible. Besides, the superposition between the second and the first clause is a simplification of the second premise, so that $\mathsf{f}(\mathsf{c}) \approx \mathsf{d}$ can be deleted from the set of clauses.

If we use relational encoding of partial functions, then N is turned into

$$\begin{gathered} \mathsf{r}(\mathsf{c},\mathsf{d}) \\ \mathsf{r}(\mathsf{c},\mathsf{e}) \\ \neg\, \mathsf{r}(x,y) \;\vee\; \neg\, \mathsf{r}(x,y') \;\vee\; y \approx y' \\ \mathsf{d} \not\approx \mathsf{e} \end{gathered}$$

In contrast to partial superposition, where we had exactly one way to derive $\mathsf{d} \approx \mathsf{e}$, there are now two different hyperresolution inferences that produce this clause, plus two further hyperresolution inferences that produce the tautologies $\mathsf{d} \approx \mathsf{d}$ and $\mathsf{e} \approx \mathsf{e}$. Moreover, we need now one further computation step to see that $\mathsf{d} \approx \mathsf{e}$ and $\mathsf{r}(\mathsf{c},\mathsf{e})$ make $\mathsf{r}(\mathsf{c},\mathsf{d})$ redundant.

We now show that the partial superposition calculus is modular for combinations of theories where all total functions are in the intersection of their signatures. Assume that we have two signatures Σ_1 and Σ_2. Call an inference pure if its premises are either all clauses over Σ_1 or they are all clauses over Σ_2. Note that a pure inference of the partial superposition calculus, in particular, derives a pure Σ_1-clause or a pure Σ_2-clause.

Theorem 29. *Suppose that Σ_1 and Σ_2 are two signatures that share the set of total function symbols and have disjoint sets of partial function symbols. Let N be a set of clauses, such that every clause in N is either a pure Σ_1-clause or a pure Σ_2-clause. Then all inferences of the partial superposition calculus with premises in N are pure.*

Proof. For the inference rules with only one premise, the result is trivial, since the clauses in N are pure. For the binary inference rules there are two possibilities: Either the term t (or t_1) in the first premise contains a partial symbol; then this symbol must also occur in the second premise so that both premises are pure clauses over the same Σ_i. Or t is an Ω^{T}-term. Since an Ω^{T}-term is smaller than every term containing a symbol from Ω^{P}, this implies that the first premise contains only total symbols, hence is both a Σ_1- and a Σ_2-clause. Again, the inference is pure. $\square$

A generalization of this result is possible if the sorts of Σ_1 and Σ_2 are taken into account: We can permit non-shared total function symbols (i. e., symbols not in $\Omega_1^{\mathrm{T}} \cap \Omega_2^{\mathrm{T}}$), provided that all these symbols have non-shared codomains (i. e., sorts not in $S_1 \cap S_2$).

Theorem 30. *Let $\Sigma_1 = (S_1, \Omega_1^{\mathrm{T}}, \Omega_1^{\mathrm{P}})$ and $\Sigma_2 = (S_2, \Omega_2^{\mathrm{T}}, \Omega_2^{\mathrm{P}})$ be two signatures such that $\Omega_1^{\mathrm{P}} \cap \Omega_2^{\mathrm{P}} = \emptyset$ and every function symbol in $\Omega_1^{\mathrm{T}} \setminus \Omega_2^{\mathrm{T}}$ (or $\Omega_2^{\mathrm{T}} \setminus \Omega_1^{\mathrm{T}}$) has a codomain in $S_1 \setminus S_2$ (or $S_2 \setminus S_1$). Let N be a set of clauses, such that every clause in N is either a pure Σ_1-clause or a pure Σ_2-clause. Let $\succ$ be a reduction ordering that is total on ground terms and that has the property that every ground term over $\Omega_1^{\mathrm{T}} \cap \Omega_2^{\mathrm{T}}$ is smaller than every ground term containing a symbol from $(\Omega_1^{\mathrm{T}} \setminus \Omega_2^{\mathrm{T}}) \cup (\Omega_2^{\mathrm{T}} \setminus \Omega_1^{\mathrm{T}})$, and every ground term over $\Omega_1^{\mathrm{T}} \cup \Omega_2^{\mathrm{T}}$ is smaller than every ground term containing a symbol from $\Omega_1^{\mathrm{P}} \cup \Omega_2^{\mathrm{P}}$.[11] Then all inferences of the partial superposition calculus with premises in N are pure.*

Proof. From the conditions on the codomains of non-shared total function symbols we can conclude that, if $f : \xi_1 \ldots \xi_n \to \xi_0 \in \Omega_1^{\mathrm{T}} \cup \Omega_2^{\mathrm{T}}$ and $\xi_0 \in S_1 \cap S_2$, then $f \in \Omega_1^{\mathrm{T}} \cap \Omega_2^{\mathrm{T}}$ and $\xi_i \in S_1 \cap S_2$ for $1 \leq i \leq n$. Consequently, every ground term that has a sort from $S_1 \cap S_2$ and does not contain partial function symbols can only consist of function symbols in $\Omega_1^{\mathrm{T}} \cap \Omega_2^{\mathrm{T}}$. It is therefore smaller than every ground term containing a non-shared function symbol. Since we are interested in total ground instances only, this implies that a variable $x : \xi$ with $\xi \in S_1 \cap S_2$ may be considered as smaller than every term containing a non-shared function symbol and every variable $y : \xi'$ with $\xi' \in (S_1 \setminus S_2) \cup (S_2 \setminus S_1)$.

With these considerations in mind, we can now proceed as in the proof of the previous theorem: Inferences with only one premise are trivially pure. For a binary inference, there are three possibilities: First, the term t or t_1 in the first premise can contain a partial symbol. Then this symbol must also occur in the second premise, so both premises are pure clauses over the same Σ_i. Second, t may contain a total symbol from $(\Omega_1^{\mathrm{T}} \setminus \Omega_2^{\mathrm{T}}) \cup (\Omega_2^{\mathrm{T}} \setminus \Omega_1^{\mathrm{T}})$ or a variable with a sort in $(S_1 \setminus S_2) \cup (S_2 \setminus S_1)$. Then such a symbol or such a variable must also occur in the term in the second premise that is unified with t. Again, both premises are pure clauses over the same Σ_i. Third, t is a term that consists exclusively

[11] For instance, a lexicographic path ordering where symbols from $\Omega_1^{\mathrm{T}} \cap \Omega_2^{\mathrm{T}}$ have lowest precedence, followed by the symbols from $(\Omega_1^{\mathrm{T}} \setminus \Omega_2^{\mathrm{T}}) \cup (\Omega_2^{\mathrm{T}} \setminus \Omega_1^{\mathrm{T}})$, followed by the symbols from $\Omega_1^{\mathrm{P}} \cup \Omega_2^{\mathrm{P}}$.

of function symbols in $\Omega_1^{\mathrm{T}} \cap \Omega_2^{\mathrm{T}}$ and variables of sorts in $S_1 \cap S_2$. Then by the properties of the ordering the first premise contains only total symbols, hence is both a Σ_1- and a Σ_2-clause, and the inference is again pure. $\square$

Example 31. Let $\Sigma_0 = (S_0, \Omega_0^{\mathrm{T}}, \emptyset)$ be the signature of a data type, where $S_0 = \{\mathsf{data}\}$ and $\Omega_0^{\mathrm{T}} = \{\mathsf{b} : \to \mathsf{data};\ \mathsf{c} : \to \mathsf{data};\ \mathsf{f} : \mathsf{data} \to \mathsf{data}\}$. We extend Σ_0 in two directions: to lists over data and to labelled trees over data: Let $S_1 = S_0 \cup \{\mathsf{list}\}$, $\Omega_1^{\mathrm{T}} = \Omega_0^{\mathrm{T}} \cup \{\mathsf{cons} : \mathsf{data}, \mathsf{list} \to \mathsf{list};\ \mathsf{nil} : \to \mathsf{list}\}$, $\Omega_1^{\mathrm{P}} = \{\mathsf{car} : \mathsf{list} \to \mathsf{data};\ \mathsf{cdr} : \mathsf{list} \to \mathsf{list}\}$. Let $S_2 = S_0 \cup \{\mathsf{tree}\}, \Omega_2^{\mathrm{T}} = \Omega_0^{\mathrm{T}} \cup \{\mathsf{treecons} : \mathsf{tree}, \mathsf{data}, \mathsf{tree} \to \mathsf{tree};\ \mathsf{empty} : \to \mathsf{tree};\ \mathsf{d} : \to \mathsf{tree}\}$, $\Omega_2^{\mathrm{P}} = \{\mathsf{label} : \mathsf{tree} \to \mathsf{data};\ \mathsf{left} : \mathsf{tree} \to \mathsf{tree};\ \mathsf{right} : \mathsf{tree} \to \mathsf{tree}\}$. [12]

Since there are no shared partial symbols in Ω_1^{P} and Ω_2^{P} and since all non-shared total symbols in Ω_1^{T} and Ω_2^{T} have non-shared codomains, the sort conditions of Thm. 30 are satisfied. If we choose an appropriate term ordering, then all inferences of the partial superposition calculus starting with a set of pure Σ_1- and pure Σ_2-clauses are pure. For instance, given the set of clauses

$$\mathsf{f}(\mathsf{f}(x)) \approx \mathsf{f}(x) \tag{1}$$

$$\mathsf{car}(\mathsf{cons}(x, l)) \approx x \tag{2}$$
$$\mathsf{cdr}(\mathsf{cons}(x, l)) \approx l \tag{3}$$
$$\mathsf{cons}(\mathsf{car}(l), \mathsf{cdr}(l)) \approx l \tag{4}$$

$$\mathsf{left}(\mathsf{treecons}(t_1, x, t_2)) \approx t_1 \tag{5}$$
$$\mathsf{label}(\mathsf{treecons}(t_1, x, t_2)) \approx x \tag{6}$$
$$\mathsf{right}(\mathsf{treecons}(t_1, x, t_2)) \approx t_2 \tag{7}$$
$$\mathsf{treecons}(\mathsf{left}(t), \mathsf{label}(t), \mathsf{right}(t)) \approx t \tag{8}$$

$$\mathsf{cons}(\mathsf{f}(x), \mathsf{cons}(x, \mathsf{nil})) \not\approx \mathsf{cons}(\mathsf{b}, \mathsf{cons}(\mathsf{b}, \mathsf{nil})) \tag{9}$$
$$\mathsf{f}(\mathsf{label}(\mathsf{d})) \approx \mathsf{b} \tag{10}$$
$$\mathsf{treecons}(\mathsf{empty}, \mathsf{c}, \mathsf{empty}) \approx \mathsf{d} \tag{11}$$

(with implicitly universally quantified variables x, y, l, t, t_1, t_2 of appropriate sorts) there is no inference between the Σ_2-clause (10) and the Σ_1-clause (9) (in contrast to the traditional superposition calculus).

The refutation proceeds as follows: Superposition of (11) and (6) yields

$$\mathsf{label}(\mathsf{d}) \approx \mathsf{c} \tag{12}$$

Superposition of (12) and (10) yields

$$\mathsf{f}(\mathsf{c}) \approx \mathsf{b} \tag{13}$$

[12] The signatures contain also operators resulting from skolemization of the problem formulas, such as the constants b, c, d.

Superposition of (13) and (1) yields

$$\mathsf{f}(\mathsf{c}) \approx \mathsf{f}(\mathsf{b}) \tag{14}$$

Superposition of (13) and (14) yields

$$\mathsf{f}(\mathsf{b}) \approx \mathsf{b} \tag{15}$$

Superposition of (15) and (9) yields

$$\mathsf{cons}(\mathsf{b}, \mathsf{cons}(\mathsf{b}, \mathsf{nil})) \not\approx \mathsf{cons}(\mathsf{b}, \mathsf{cons}(\mathsf{b}, \mathsf{nil})) \tag{16}$$

from which equality resolution derives the empty clause.

Note that both the Σ_1- and the Σ_2-"module" of the prover have to perform inferences with Σ_0-clauses (and possibly even inferences that involve only Σ_0-clauses). This is a significant difference to the calculus for hierarchic structures described in the next chapter, where reasoning with formulas over the common vocabulary is completely left to one of the two deduction modules.

4 Hierarchic Extensions

The inference system of the partial superposition calculus (and its completeness proof) can be turned – with slight modifications – into a calculus for hierarchic structures.

Definition 32. A signature $\Sigma_1 = (S_1, \Omega_1^{\mathrm{T}}, \Omega_1^{\mathrm{P}})$ is called an *extension* of a signature $\Sigma_0 = (S_0, \Omega_0^{\mathrm{T}}, \Omega_0^{\mathrm{P}})$ if $S_1 \supseteq S_0$, $\Omega_1^{\mathrm{T}} \supseteq \Omega_0^{\mathrm{T}}$ and $\Omega_1^{\mathrm{P}} \supseteq \Omega_0^{\mathrm{P}}$.

Definition 33. Let $\Sigma_1 = (S_1, \Omega_1^{\mathrm{T}}, \Omega_1^{\mathrm{P}})$ be an extension of $\Sigma_0 = (S_0, \Omega_0^{\mathrm{T}}, \Omega_0^{\mathrm{P}})$; let A be a Σ_1-algebra. The Σ_0*-reduct* of A, denoted by $A|_{\Sigma_0}$, is the Σ_0-algebra that is obtained from A by removing all sets ξ_A for $\xi \in S_1 \setminus S_0$ and all functions f_A, g_A for $f \in (\Omega_1^{\mathrm{T}} \setminus \Omega_0^{\mathrm{T}})$, $g \in (\Omega_1^{\mathrm{P}} \setminus \Omega_0^{\mathrm{P}})$.

Convention 34. In the rest of this paper, we will only consider signature extensions where $\Omega_0^{\mathrm{P}} = \emptyset$ and all symbols in $\Omega_1^{\mathrm{T}} \setminus \Omega_0^{\mathrm{T}}$ have a codomain in $S_1 \setminus S_0$.[13]

Definition 35. A *universal Σ-theory* $\mathcal{T}$ is a set of universally quantified Σ-formulae.

[13] If relation symbols are encoded as functions as described in footnote 2, one can choose a new sort for every relation symbol. Hence extensions by new relation symbols are not restricted.

Definition 36. Let $\Sigma_1 = (S_1, \Omega_1^{\mathrm{T}}, \Omega_1^{\mathrm{P}})$ be an extension of $\Sigma_0 = (S_0, \Omega_0^{\mathrm{T}}, \emptyset)$; let $\mathcal{T}_0$ be a universal Σ_0-theory. A Σ_1-formula F *follows from a set* N *of* Σ_1-*formulae relative to* $\mathcal{T}_0$ (denoted by $N \models_{\mathcal{T}_0} F$) if every model of N whose reduct to Σ_0 is a model of $\mathcal{T}_0$ is also a model of F. [14]

We assume the following scenario: Let $\Sigma_0 = (S_0, \Omega_0^{\mathrm{T}}, \emptyset)$ be a (total) signature, and let $\mathcal{T}_0$ be some universal Σ_0-theory for which we have a refutationally complete prover (or even a decision procedure) that is able to check the unsatisfiability of sets of Σ_0-clauses w. r. t. $\mathcal{T}_0$. Let $\Sigma_1 = (S_1, \Omega_1^{\mathrm{T}}, \Omega_1^{\mathrm{P}})$ be an extension of Σ_0 such that all symbols in $\Omega_1^{\mathrm{T}} \setminus \Omega_0^{\mathrm{T}}$ have a codomain in $S_1 \setminus S_0$. Let N be a set of Σ_1-clauses. The task is to check whether N is unsatisfiable relative to $\mathcal{T}_0$, that is, whether $N \models_{\mathcal{T}_0} \bot$, using the prover for $\mathcal{T}_0$ as a black-box. To this end, we will modify the rules of the partial superposition calculus as follows:

- The inference rules are applied to clauses where non-variable Σ_0-terms have been "abstracted out" (see below). [15]
- A new inference rule is introduced that allows us to derive a contradiction from any finite set of Σ_0-clauses that is inconsistent with $\mathcal{T}_0$.
- Since the Σ_0-part is left to the $\mathcal{T}_0$-prover, none of the old inference rules are applied if inferences involve only Σ_0-terms.

From an operational point of view, it is usually advisable to use an incremental $\mathcal{T}_0$-prover (or, at a pinch, several instances of a non-incremental $\mathcal{T}_0$-prover) that runs in parallel with the main prover and receives all base clauses that are generated by the the main prover. Classes of clause sets for which saturation under the old rules is known to terminate are an exception – here the $\mathcal{T}_0$-prover can be called when the main prover has terminated.

We write $N \models_{\mathcal{T}_0}^{\mathrm{TG}} F$ if every total-term-generated model of N whose Σ_0-reduct is a model of $\mathcal{T}_0$ is also a model of F. For refutational theorem proving, $\models_{\mathcal{T}_0}$ can be replaced by $\models_{\mathcal{T}_0}^{\mathrm{TG}}$:

Proposition 37. *Let N be a set of universally quantified clauses. Then $N \models_{\mathcal{T}_0} \bot$ if and only if $N \models_{\mathcal{T}_0}^{\mathrm{TG}} \bot$.*

Proof. Analogously to the proof of Lemma 9. □

[14] The results of this section hold also if one considers an arbitrary compact set $\mathcal{C}$ of term-generated Σ_0-algebras (closed under isomorphism) instead of a universal Σ_0-theory $\mathcal{T}_0$. In this case, F follows from N relative to $\mathcal{C}$ if every model of N whose Σ_0-reduct is contained in $\mathcal{C}$ is also a model of F.

[15] Instead of abstracting out non-variable Σ_0-terms eagerly, one can also treat non-variable Σ_0-subterms in the unification algorithm in a similar way as in Digricoli and Harrison's RUE rule [9], i. e., by turning an appropriate disagreement set into a list of negative literals. We have done this in a previous version [14] of this paper.

We call sorts from S_0 *base sorts* and sorts from $S_1 \setminus S_0$ *extension sorts*; analogously, a function symbol from Ω_0^{T} is called *base symbol* and a function symbol from $\Omega_1^{\mathrm{P}} \cup \Omega_1^{\mathrm{T}} \setminus \Omega_0^{\mathrm{T}}$ is called *extension symbol.* A term is called a *base term* if it consists only of base symbols and variables of base sorts; it is called an *extension term* if it contains at least one extension symbol or variable of an extension sort.

From the conditions on the codomains of total extension symbols we can conclude that, if $f : \xi_1 \dots \xi_n \to \xi_0 \in \Omega_1^{\mathrm{T}}$ and $\xi_0 \in S_0$, then $f \in \Omega_0^{\mathrm{T}}$ and $\xi_i \in S_0$ for $1 \leq i \leq n$. Consequently, every ground term that has a base sort and does not contain partial function symbols must be a base term.

An extension term is called *abstracted* if it has no non-variable base subterms. A literal $t \mathrel{\dot{\approx}} t'$ is called a *base literal* if both t and t' are base terms; it is called an *abstracted extension literal* if one of the two terms is an abstracted extension term and the other one is an abstracted extension term or a variable.

A clause is called *abstracted* if all its literals are either base literals or abstracted extension literals. Every clause C can be transformed into an equivalent abstracted clause in the following way: whenever a non-variable base term t occurs immediately below an extension symbol, then it is replaced by a new variable x (or "abstracted out") and the literal $x \not\approx t$ is added to C. This transformation is repeated until all non-variable base terms below extension symbols have been eliminated, then the abstraction operation is applied to non-variable base terms occurring in equations with extension terms. The resulting clause is denoted by $\mathrm{abs}(C)$. [16]

Lemma 38. *Let C be a clause, let A be an algebra. Then $A \models C$ if and only if $A \models \mathrm{abs}(C)$.*

Proof. If is sufficient to show that, for every A and β, $(A, \beta)(C[t]) \geq \frac{1}{2}$ if and only if $(A, \beta)(\forall x.(x \not\approx t \vee C[x])) \geq \frac{1}{2}$, where t is a base term and x does not occur in C. For the "if" part, assume that $\frac{1}{2} \leq (A, \beta)(\forall x.(x \not\approx t \vee C[x]))$, then $\frac{1}{2} \leq \min \{\, (A, \beta[x \mapsto a])(x \not\approx t \vee C[x]) \mid a \in \xi_A \,\} \leq (A, \beta[x \mapsto (A, \beta)(t)])(x \not\approx t \vee C[x]) = (A, \beta)(C[t])$. For the "only if" part, assume that $\frac{1}{2} \leq (A, \beta)(C[t])$. If $(A, \beta)(t) = a$, then clearly $(A, \beta[x \mapsto a])(x \not\approx t \vee C[x]) = (A, \beta[x \mapsto a])(C[x]) = (A, \beta)(C[t]) \geq \frac{1}{2}$. Otherwise, $(A, \beta[x \mapsto a])(x \not\approx t \vee C[x]) \geq (A, \beta[x \mapsto a])(x \not\approx t) = 1$. $\square$

We assume that all input clauses are transformed into equivalent abstracted clauses before we start the saturation process. Most of the inference rules of the partial superposition calculus preserve abstraction. *Superposition* and

[16] Note that we abstract out only base terms. Abstracting out terms that contain *partial* function symbols would *not* yield an equivalent clause. For instance, if g is a partial function symbol, then $\mathsf{g}(\mathsf{c}) \approx \mathsf{c}$ does not follow from $x \not\approx \mathsf{g}(\mathsf{c}) \vee x \approx \mathsf{c}$.

merging paramodulation are the exceptions: for these rules we have to perform abstraction on the conclusion explicitly.

Inference System 39. The inference system of the *constraint partial superposition calculus* (CPS, for short) consists of the inference rules *equality resolution*, *superposition*, *partial top-superposition*, *merging paramodulation*, *factoring*, and *constraint refutation*.

The CPS calculus is parameterized by a reduction ordering $\succ$ on terms that is total on ground terms and that has the property that every ground term over Ω_0^{T} is smaller than every ground term containing a symbol from $\Omega_1^{\mathrm{P}} \cup \Omega_1^{\mathrm{T}} \setminus \Omega_0^{\mathrm{T}}$ and every ground term over Ω_1^{T} is smaller than every ground term containing a symbol from Ω_1^{P} (for instance, a lexicographic path ordering where symbols from Ω_0^{T} have lowest precedence, followed by the symbols from Ω_1^{T}, followed by the symbols from Ω_1^{P}).[17] The extension to a literal and clause ordering is defined as before.

A literal that is involved in an inference must be maximal in the respective clause (except for the literal $s_0 \approx s_0'$ in *merging paramodulation* and the literals $t_i \approx t_i'$ $(i > 1)$ in *partial top-superposition*). A positive literal that is involved in a *superposition*, *partial top-superposition*, or *merging paramodulation* inference must be strictly maximal in the respective clause (with the exceptions above). Except for the *constraint refutation* rule, a literal that is involved in an inference must be an abstracted extension literal. In inferences with two premises, the left premise is not greater than or equal to the right premise.

Equality Resolution
$$\frac{C' \vee s \not\approx s'}{C'\sigma}$$
if s does not contain partial function symbols and σ is a total most general unifier of s and s'.[18]

Superposition
$$\frac{D' \vee t \approx t' \qquad C' \vee s[u] \,\dot{\approx}\, s'}{\operatorname{abs}\big((D' \vee C' \vee s[t'] \,\dot{\approx}\, s')\sigma\big)}$$
if u is not a variable, t does not contain partial function symbols below the top, σ is a total most general unifier of t and u, $t\sigma \not\preceq t'\sigma$, $s\sigma \not\prec s'\sigma$, and, if $s \,\dot{\approx}\, s'$ occurs positively or s contains no partial function symbols, then $s\sigma \not\preceq s'\sigma$.

[17] Since we are interested in total ground instances only, this implies that a variable of base sort may be considered as smaller than every term containing an extension symbol or variable of an extension sort.

[18] By the global requirement that $s \not\approx s'$ is an abstracted extension literal and the sort condition of Convention 34, this implies that both s and s' have an extension sort and consist only of variables and total extension symbols.

Partial Top-Superposition
$$\frac{D' \lor t_1 \approx t'_1 \lor \ldots \lor t_n \approx t'_n \qquad C' \lor s \approx s'}{(D' \lor C' \lor s' \approx t'_1 \lor \ldots \lor s' \approx t'_n)\sigma}$$

if $n \geq 2$, s contains a partial function symbol at the top and no partial function symbols below the top, each t'_i contains a partial function symbol, σ is a total most general unifier of s and all t_i, $t_i\sigma \not\preceq t'_i\sigma$, $s\sigma \not\preceq s'\sigma$, and $s'\sigma \not\preceq t'_i\sigma$.

Merging Paramodulation
$$\frac{D' \lor t \approx t' \qquad C' \lor s_0 \approx s'_0 \lor s \approx s'[u]}{\mathrm{abs}\Big((D' \lor C' \lor s_0 \approx s'_0 \lor s \approx s'[t'])\sigma\Big)}$$

if u is not a variable, t does not contain partial function symbols below the top, σ is a total most general simultaneous unifier of t and u and of s_0 and s, $t\sigma \not\preceq t'\sigma$, $s\sigma \not\preceq s'\sigma$, $s\sigma \not\preceq s'_0\sigma$, and $s'\sigma \not\preceq s'_0\sigma$,

Factoring
$$\frac{C' \lor s \approx s' \lor t \approx t'}{(C' \lor s \approx s')\sigma}$$

if σ is a total most general simultaneous unifier of s and t and of s' and t'.

Constraint Refutation
$$\frac{M}{\bot}$$

if M is a finite set of Σ_0-clauses that is inconsistent with the base theory $\mathcal{T}_0$, that is, $M \models_{\mathcal{T}_0} \bot$.

Theorem 40. *The inference rules of the constraint partial superposition calculus are sound w. r. t. $\models_{\mathcal{T}_0}$ (and therefore also sound w. r. t. $\models^{\mathrm{TG}}_{\mathcal{T}_0}$).*

Proof. Analogously to the proof of Thm. 16. □

To define a redundancy criterion for the CPS calculus and to show its refutational completeness, we use the concept of approximation introduced by Bachmair, Ganzinger, and Waldmann [5].

The following definition relates inferences of the constraint partial superposition calculus to ground inferences of the partial superposition calculus. Note that the explicit abstraction in the *superposition* and *merging paramodulation* rules of the constraint partial superposition calculus produces additional negative base literals and that we have to cater to them.

Definition 41. Let ι be an inference of the CPS calculus with abstracted premises $C_1, \ldots, C_n$ and conclusion C. Let ι' be an inference of the partial superposition calculus with ground premises $C'_1, \ldots, C'_n$ and conclusion C'. We say that ι' is a *total ground instance* of ι if σ is a total substitution,

$C_i\sigma = C_i'$, and $C\sigma = C' \vee C''$, where all literals in C'' have the form $t \not\approx t$ for a base term t. The set of all total ground instances of ι is denoted by $\mathrm{tgi}(\iota)$.

Definition 42. Let N be a set of abstracted clauses. We define $\mathit{Red}^{\mathrm{C}}_{\mathrm{CPS}}(N)$ as the set of all abstracted clauses C such that $\mathrm{tgi}(C) \subseteq \mathit{Red}^{\mathrm{C}}_{\mathrm{PSG}}(\mathrm{tgi}(N))$. We define $\mathit{Red}^{\mathrm{I}}_{\mathrm{CPS}}(N)$ as the set of all inferences ι of the CPS calculus such that either ι is a *constraint refutation* inference and $\bot \in N$, or ι is not a *constraint refutation* inference and $\mathrm{tgi}(\iota) \subseteq \mathit{Red}^{\mathrm{I}}_{\mathrm{PSG}}(\mathrm{tgi}(N))$.

Lemma 43. *The pair* $(\mathit{Red}^{\mathrm{I}}_{\mathrm{CPS}}, \mathit{Red}^{\mathrm{C}}_{\mathrm{CPS}})$ *is a redundancy criterion with respect to* $\models^{\mathrm{TG}}_{\mathcal{T}_0}$.

Proof. Analogously to the proof of Lemma 22. □

Let A be a term-generated Σ_0-model of $\mathcal{T}_0$. For every ground base term t let $m(t)$ be the smallest ground base term of the congruence class of t in A. We define a rewrite system Eq'_A by $\mathrm{Eq}'_A = \{\, t \to m(t) \mid t \in \mathrm{T}_{\Sigma_0}(\emptyset),\, t \neq m(t) \,\}$. Obviously, Eq'_A is terminating, right-reduced, and confluent. Now let Eq_A be the set of all rules $l \to r$ in Eq'_A such that l is not reducible by $\mathrm{Eq}'_A \setminus \{l \to r\}$. It is fairly easy to prove that Eq'_A and Eq_A define the same set of normal forms, and from this we can conclude that Eq_A and Eq'_A induce the same equality relation on $\mathrm{T}_{\Sigma_0}(\emptyset)$. We identify Eq_A with the set of clauses $\{\, t \approx t' \mid t \to t' \in \mathrm{Eq}_A \,\}$. Let Deq_A be the set of all clauses $t \not\approx t'$, such that t and t' are distinct ground base terms in normal form with respect to Eq_A.

Lemma 44. *Let A be a term-generated Σ_0-model of $\mathcal{T}_0$ and let C be a ground Σ_0-clause. Then C is true in A if and only if there exist clauses $C_1, \ldots, C_n$ in* $\mathrm{Eq}_A \cup \mathrm{Deq}_A$ *such that $C_1, \ldots, C_n \models C$ and $C \succeq_{\mathrm{C}} C_i$ for $1 \leq i \leq n$.*

Proof. The "if" part follows from the fact that all clauses in $\mathrm{Eq}_A \cup \mathrm{Deq}_A$ are true in A. For the "only if" part observe that a ground Σ_0-clause C is true in A if and only if one of its literals is true in A. If this is a positive literal $s \approx s'$, take those clauses in Eq_A that are used to rewrite s and s' to the same normal form; if it is a negative literal $s \not\approx s'$, take those clauses in Eq_A that are used to rewrite s and s' to their normal forms t and t' plus the clause $t \not\approx t' \in \mathrm{Deq}_A$. □

Let N be a set of abstracted Σ_1-clauses and A a term-generated Σ_0-model of $\mathcal{T}_0$, then N_A denotes the set $\mathrm{Eq}_A \cup \mathrm{Deq}_A \cup \{\, C\sigma \mid C \in N,\, \sigma$ total and reduced with respect to $\mathrm{Eq}_A,\, C\sigma$ ground$\,\}$.

Lemma 45. *If N is a set of abstracted clauses, then we have* $\mathit{Red}^{\mathrm{I}}_{\mathrm{PSG}}(\mathrm{tgi}(N)) \subseteq \mathit{Red}^{\mathrm{I}}_{\mathrm{PSG}}(N_A)$.

Proof. Obviously $\mathit{Red}^{\mathrm{I}}_{\mathrm{PSG}}(\mathrm{tgi}(N)) \subseteq \mathit{Red}^{\mathrm{I}}_{\mathrm{PSG}}(\mathrm{Eq}_A \cup \mathrm{Deq}_A \cup \mathrm{tgi}(N))$. Let C be a clause in $\mathrm{Eq}_A \cup \mathrm{Deq}_A \cup \mathrm{tgi}(N)$ and not in N_A. As $C = C'\sigma$ for some $C' \in N$, it follows from $C'\rho$ and $\mathrm{Eq}_A \cup \mathrm{Deq}_A$, where ρ is the substitution that maps every variable x to the Eq_A-normal form of $x\sigma$. Since C follows from smaller clauses in $\mathrm{Eq}_A \cup \mathrm{Deq}_A \cup \mathrm{tgi}(N)$, it is in $\mathit{Red}^{\mathrm{C}}_{\mathrm{PSG}}(\mathrm{Eq}_A \cup \mathrm{Deq}_A \cup \mathrm{tgi}(N))$. Hence $\mathit{Red}^{\mathrm{I}}_{\mathrm{PSG}}(\mathrm{Eq}_A \cup \mathrm{Deq}_A \cup \mathrm{tgi}(N)) \subseteq \mathit{Red}^{\mathrm{I}}_{\mathrm{PSG}}(N_A)$. □

Proposition 46. *Let A be a term-generated Σ_0-model of $\mathcal{T}_0$ and let N be a set of abstracted clauses. If A satisfies all Σ_0-clauses in N and N is saturated w. r. t. the CPS calculus and $(\mathit{Red}^{\mathrm{I}}_{\mathrm{CPS}}, \mathit{Red}^{\mathrm{C}}_{\mathrm{CPS}})$, then N_A is saturated w. r. t. the partial superposition calculus and $(\mathit{Red}^{\mathrm{I}}_{\mathrm{PSG}}, \mathit{Red}^{\mathrm{C}}_{\mathrm{PSG}})$.*

Proof. We have to show that every inference from clauses of N_A is redundant with respect to N_A, i. e., that it is contained in $\mathit{Red}^{\mathrm{I}}_{\mathrm{PSG}}(N_A)$. We demonstrate this in detail for the *equality resolution* and the *superposition* rule. The analysis of the other rules is similar. Note that by Lemma 44 every base clause that is true in A and is not contained in $\mathrm{Eq}_A \cup \mathrm{Deq}_A$ follows from smaller clauses in $\mathrm{Eq}_A \cup \mathrm{Deq}_A$, thus it is in $\mathit{Red}^{\mathrm{C}}_{\mathrm{PSG}}(N_A)$; every inference involving such a clause is in $\mathit{Red}^{\mathrm{I}}_{\mathrm{PSG}}(N_A)$.

The *equality resolution* rule is obviously not applicable to clauses from $\mathrm{Eq}_A \cup \mathrm{Deq}_A$. Suppose that ι is an *equality resolution* inference with a premise $C\sigma$, where $C \in N$ and σ is a total substitution and reduced with respect to Eq_A. If C is a base clause, then ι is in $\mathit{Red}^{\mathrm{I}}_{\mathrm{PSG}}(N_A)$. If C is not a base clause, then ι is a total ground instance of an inference ι' from C. Since ι' is in $\mathit{Red}^{\mathrm{I}}_{\mathrm{CPS}}(N)$, ι is in $\mathit{Red}^{\mathrm{I}}_{\mathrm{PSG}}(\mathrm{tgi}(N))$, again this implies $\iota \in \mathit{Red}^{\mathrm{I}}_{\mathrm{PSG}}(N_A)$.

Obviously a clause from Deq_A cannot be the first premise of a *superposition* inference. Suppose that the first premise is a clause from Eq_A. The second premise cannot be a non-base clause, since all ground terms in the substitution part of a clause $C\sigma$ are reduced; as it is a base clause, the inference is redundant. Now suppose that ι is a *superposition* inference with a first premise $C\sigma$, where $C \in N$ and σ is a total substitution and reduced with respect to Eq_A. If C is a base clause, then ι is in $\mathit{Red}^{\mathrm{I}}_{\mathrm{PSG}}(N_A)$. Otherwise, we can conclude that the second premise can be written as $C'\sigma$, where $C' \in N$ is not a base clause (without loss of generality, C and C' do not have common variables). We have to distinguish between two cases: If the overlap takes place below a variable occurrence, the conclusion of the inference follows from $C\sigma$ and some instance $C'\rho$, which are both smaller than $C'\sigma$. Otherwise, ι is a total ground instance of an inference ι' from C. In both cases, ι is contained in $\mathit{Red}^{\mathrm{I}}_{\mathrm{PSG}}(N_A)$. □

Theorem 47. *Let N be a set of clauses that is saturated w. r. t. the CPS calculus. Then N has a model whose Σ_0-reduct is a model of the base theory $\mathcal{T}_0$ if and only if N does not contain the empty clause.*

Proof. If N contains the empty clause, then obviously it does not have a model.

If $\bot \notin N$, then we can first show that there is a Σ_0-algebra that is a Σ_0-model of all Σ_0-clauses in N and of the base theory $\mathcal{T}_0$: Assume otherwise. Then, by compactness of first-order logic, some finite subset of the Σ_0-clauses in N must be inconsistent with $\mathcal{T}_0$, hence the *constraint refutation* rule is applicable to this subset. By saturation, this inference must be redundant. But that is only possible if $\bot \in N$, contradicting our assumption.

Now let A be some Σ_0-model of the Σ_0-clauses in N and of $\mathcal{T}_0$. Since both N and $\mathcal{T}_0$ consist only of universally quantified formulae, we may assume without loss of generality that A is term-generated. By Prop. 46, the set N_A is saturated w. r. t. the partial superposition calculus and $(Red^{\mathrm{I}}_{\mathrm{PSG}}, Red^{\mathrm{C}}_{\mathrm{PSG}})$. Clearly, $\bot \notin N_A$, so N_A has a total-term-generated model A'. Since $N_A \models \mathrm{tgi}(N)$, A' is also a model of $\mathrm{tgi}(N)$ and therefore a model of N. Furthermore, since A' is a model of $\mathrm{Eq}_A \cup \mathrm{Deq}_A$ and total-term-generated, the Σ_0-reduct of A' is isomorphic to A and therefore a model of $\mathcal{T}_0$. □

Example 48. Let $\Sigma_0 = (S_0, \Omega_0^{\mathrm{T}}, \emptyset)$ be the signature of a data type, where $S_0 = \{\mathsf{data}\}$ and $\Omega_0^{\mathrm{T}} = \{\mathsf{b} : \to \mathsf{data};\ \mathsf{c} : \to \mathsf{data};\ \mathsf{f} : \mathsf{data} \to \mathsf{data}\}$; let $\mathcal{T}_0$ be the theory $\{\forall x. \mathsf{f}(\mathsf{f}(x)) \approx \mathsf{f}(x)\}$.

We extend Σ_0 to lists over data: Let $S_1 = S_0 \cup \{\mathsf{list}\}$, $\Omega_1^{\mathrm{T}} = \Omega_0^{\mathrm{T}} \cup \{\mathsf{cons} : \mathsf{data}, \mathsf{list} \to \mathsf{list};\ \mathsf{nil} : \to \mathsf{list};\ \mathsf{d} : \to \mathsf{list}\}$, $\Omega_1^{\mathrm{P}} = \{\mathsf{car} : \mathsf{list} \to \mathsf{data};\ \mathsf{cdr} : \mathsf{list} \to \mathsf{list}\}$.[19]

Consider the following set of clauses:

$$\mathsf{car}(\mathsf{cons}(x, l)) \approx x \tag{1}$$
$$\mathsf{cdr}(\mathsf{cons}(x, l)) \approx l \tag{2}$$
$$\mathsf{cons}(\mathsf{car}(l), \mathsf{cdr}(l)) \approx l \tag{3}$$

$$\mathsf{f}(\mathsf{c}) \approx \mathsf{b} \tag{4}$$
$$\mathsf{f}(\mathsf{f}(\mathsf{b})) \not\approx \mathsf{car}(\mathsf{cdr}(\mathsf{cons}(\mathsf{f}(\mathsf{b}), \mathsf{cons}(\mathsf{b}, \mathsf{d})))) \tag{5}$$

(with implicitly universally quantified variables x, l of appropriate sorts). We will show that this set is unsatisfiable relative to $\mathcal{T}_0$ using the CPS calculus:

We replace (5) by its abstracted version

$$x \not\approx \mathsf{f}(\mathsf{f}(\mathsf{b})) \lor y \not\approx \mathsf{f}(\mathsf{b}) \lor z \not\approx \mathsf{b} \lor x \not\approx \mathsf{car}(\mathsf{cdr}(\mathsf{cons}(y, \mathsf{cons}(z, \mathsf{d})))) \tag{6}$$

Superposition of (2) and (6) yields

$$x \not\approx \mathsf{f}(\mathsf{f}(\mathsf{b})) \lor y \not\approx \mathsf{f}(\mathsf{b}) \lor z \not\approx \mathsf{b} \lor x \not\approx \mathsf{car}(\mathsf{cons}(z, \mathsf{d})) \tag{7}$$

[19] The signatures contain also operators resulting from skolemization of the problem formulas, such as the constants b, c, d.

Superposition of (1) and (7) yields

$$x \not\approx \mathsf{f}(\mathsf{f}(\mathsf{b})) \vee y \not\approx \mathsf{f}(\mathsf{b}) \vee z \not\approx \mathsf{b} \vee x \not\approx z \tag{8}$$

The base clauses (4) and (8) (the latter is actually equivalent to the ground clause $\mathsf{f}(\mathsf{f}(\mathsf{b})) \not\approx \mathsf{b}$) are inconsistent with $\mathcal{T}_0$, so constraint refutation yields the empty clause.

5 Shallow and Local Extensions of a Base Theory

As shown in Sect. 4, constraint partial superposition is complete whenever every function in the extension whose codomain is a base sort is declared as partial, whereas a function whose codomain is an extension sort can be declared as either total or partial. From our point of view, an important application of this result is to approximate refutational theorem proving in extensions of base theories for which refutationally complete black box theorem provers exist. If constraint partial superposition finds a contradiction for a set of clauses in the extended signature, the set is unsatisfiable in particular also with respect to total algebras. In that sense constraint partial superposition is a sound and modular approximation of refutational theorem proving for hierarchical first-order theories. In this section we discuss cases when this approximation is, in fact, complete. A particularly simple case is that of a shallow extension. Local extensions of theories are another case.

Let $\Sigma_0 = (S_0, \Omega_0^{\mathrm{T}}, \emptyset)$ be a (total) signature, and let $\mathcal{T}_0$ be a first-order theory over Σ_0. We will consider extensions $\Sigma_1 = (S_1, \Omega_1^{\mathrm{T}}, \Omega_1^{\mathrm{P}})$ of Σ_0, such that all symbols in $\Omega_1^{\mathrm{T}} \setminus \Omega_0^{\mathrm{T}}$ have a codomain in $S_1 \setminus S_0$, and first-order theories over such signature extensions. We say that a first-order theory $\mathcal{T}_1$ over Σ_1 is an *extension* of the theory $\mathcal{T}_0$ if $\mathcal{T}_1 = \mathcal{T}_0 \cup \mathcal{F}$, where $\mathcal{F}$ is a set of first-order formulae over Σ_1. In what follows we will consider only extensions $\mathcal{T}_0 \subseteq \mathcal{T}_1 = \mathcal{T}_0 \cup \mathcal{F}$ where $\mathcal{F}$ is a set of (universally quantified) clauses.

In the rest of this paper, we will talk both about the *partial algebra semantics* and the *total algebra semantics* of a theory. The *partial algebra semantics* refers to the notions of (partial) models, satisfiability, entailment, etc., as described in Sect. 2. Note that a Σ_1-algebra A is a (partial) model of the extension $\mathcal{T}_1$ of $\mathcal{T}_0$ if it is a model of $\mathcal{F}$ and if its Σ_0-reduct $A|_{\Sigma_0}$ is a total model of $\mathcal{T}_0$. The *total algebra semantics* is defined analogously; here we consider only total algebras, i. e., algebras where all function symbols are interpreted by total functions. For instance, a set of formulas is unsatisfiable in the total algebra semantics if it has no total model.

We consider a special class of extensions of a base theory, namely shallow extensions. These are extensions by clauses in which partial function symbols occur only in positive literals and only at the root of terms. We show that in this case every partial model can be extended to a total model and, therefore, constraint partial superposition is complete also with the total algebra semantics.

Definition 49. Suppose $\mathcal{T}_0 \subseteq \mathcal{T}_1$ is a theory extension in which all functions in the extension $\Sigma_1 \setminus \Sigma_0$ having a codomain in the set S_0 of (base) sorts in Σ_0 are declared as partial. A Σ_1-clause C is called *shallow* if partial function symbols occur in C only positively and only at the root of terms. The theory extension $\mathcal{T}_0 \subseteq \mathcal{T}_1$ is shallow if $\mathcal{T}_1 \setminus \mathcal{T}_0$ consists only of shallow clauses.

The definition of shallow terms given above is a generalization of the corresponding notion used e. g. by Comon, Haberstrau, and Jouannaud [8], Nieuwenhuis [24], or Jacquemard, Meyer and Weidenbach [19]. The difference is that we consider terms which are shallow w. r. t. a subset of the function symbols, whose elements are declared to be partial. When defining shallow clauses we require that terms containing partial function symbols only occur positively because without this requirement any set of clauses could be made shallow by using variable abstraction.

Example 50. Suppose we have the natural numbers (of sort nat) as base theory. Consider as an extension the two clauses

$$\mathsf{read}(\mathsf{write}(ar, i, x), i) \approx x$$
$$i \approx j \vee \mathsf{read}(\mathsf{write}(ar, i, x), j) \approx \mathsf{read}(ar, j)$$

where array is a new sort, $\mathsf{write} : \mathsf{array}, \mathsf{nat}, \mathsf{nat} \to \mathsf{array}$ is a total and $\mathsf{read} : \mathsf{array}, \mathsf{nat} \to \mathsf{nat}$ a partial function symbol, and ar, i, j, x are variables of suitable sort. Under these assumptions the two clauses are shallow.

This definition of read is tail-recursive, and in general tail-recursive definitions of a partial function will be shallow. Other kinds of recursive definitions will normally not be shallow, as exemplified by the case of length over lists (with the natural numbers as base theory):

$$\mathsf{length}(\mathsf{cons}(x, l)) \approx \mathsf{succ}(\mathsf{length}(l))$$

where the base function $\mathsf{succ} : \mathsf{nat} \to \mathsf{nat}$ and the extension function $\mathsf{cons} : \mathsf{nat}, \mathsf{list} \to \mathsf{list}$ are total, whereas $\mathsf{length} : \mathsf{list} \to \mathsf{nat}$ must be partial due to the sort condition of Convention 34.

Shallow extensions enjoy the property that any partial model can be extended to a total model.

Theorem 51. *Suppose that $\mathcal{T}_0 \subseteq \mathcal{T}_1$ is a theory extension in which all functions in $\Sigma_1 \setminus \Sigma_0$ with a codomain in S_0 are declared as partial. If all clauses in $\mathcal{T}_1 \setminus \mathcal{T}_0$ are shallow, then $\mathcal{T}_1$ has a partial model if and only if $\mathcal{T}_1$ has a total model.*

Proof. Suppose A is a partial Σ_1-algebra that is a model of $\mathcal{T}_1$. Pick, for each sort ξ, an element a_ξ from the carrier ξ_A associated with the sort ξ in A. Let B be the extension of A into a total algebra obtained by making f_B return a_ξ, wherever f_A is undefined in A, for every partial f of codomain ξ. It is easy to see that B is also a model of $\mathcal{T}_1$: Since all function symbols in Σ_0 are total, $B|_{\Sigma_0}$ coincides with $A|_{\Sigma_0}$ so that B is a model of $\mathcal{T}_0$. Suppose $\mathcal{T}_1 \setminus \mathcal{T}_0$ contains, say, an equation $f(\vec{s}) \approx g(\vec{t})$ where f is partial and has the codomain ξ. Since the equation is shallow, neither $\vec{s}$ nor $\vec{t}$ contain any partial function symbol. Thus, for each assignment of the variables, the values $\vec{a}$ and $\vec{b}$ for $\vec{s}$ and $\vec{t}$, respectively, are defined. Therefore, in order for the equation to be satisfied in A, f_A is defined on $\vec{a}$ if and only if g_A is defined on $\vec{b}$. If $f_A(\vec{a})$ is defined, so is $g_A(\vec{b})$, and $f_B(\vec{a}) = f_A(\vec{a}) = g_A(\vec{b}) = g_B(\vec{b})$. If $f_A(\vec{a})$ is undefined, so is $g_A(\vec{b})$, thus $f_B(\vec{a}) = a_\xi = g_B(\vec{b})$. For the case of general clauses also note that partial functions do not occur negatively in shallow clauses. $\square$

Note that any set of ground clauses can be turned into a set of shallow ground clauses by introducing new constants for subterms that start with a partial function:

Definition 52. Suppose that $\mathcal{T}_0 \subseteq \mathcal{T}_1$ is a theory extension. Let G be a set of ground clauses in the signature Σ_1. Then G_F is the set of clauses that we obtain from G if we replace in a bottom-up manner every term $g(t_1, \ldots, t_n)$ with $g \in \Omega_1^{\mathrm{P}}$ by a new (total) constant c and add the definition $g(t_1, \ldots, t_n) \approx c$ to the set of clauses. The set of new constants introduced during this process is denoted by Ω_c.

This *flattening transformation* preserves [un-]satisfiability with respect to total algebra semantics. It does not preserve satisfiability with respect to the partial algebra semantics, though, as shown by the following example:

Example 53. Let $\Omega^{\mathrm{T}} = \{\mathsf{nil}/0, \mathsf{cons}/2\}$, $\Omega^{\mathrm{P}} = \{\mathsf{car}/1, \mathsf{cdr}/1\}$, and let A be the algebra of finite lists with the usual interpretation of these symbols. In particular, we assume that $\mathsf{car}(\mathsf{nil})$ is undefined in A. Let G consist of the unit ground clause:

$$\mathsf{car}(\mathsf{nil}) \not\approx \mathsf{car}(\mathsf{nil}).$$

Let G_F be obtained from G by the flattening transformation above, i. e. by replacing the two occurrences of $\mathsf{car}(\mathsf{nil})$ by new total constants and adding the definitions to the set of clauses. G_F consists of the following clauses:

$$\mathsf{car}(\mathsf{nil}) \approx \mathsf{c}$$
$$\mathsf{car}(\mathsf{nil}) \approx \mathsf{d}$$
$$\mathsf{c} \not\approx \mathsf{d}.$$

Clearly, G is satisfiable, as $A \models \mathsf{car}(\mathsf{nil}) \not\approx \mathsf{car}(\mathsf{nil})$ since both sides are undefined in A. However, G_F is unsatisfiable, as in any model of the two clauses in G_F $\mathsf{car}(\mathsf{nil})$ must be defined and equal to both c and d, and hence the third clause cannot be true.

In what follows, if not otherwise specified, we will always assume that $\mathcal{T}_0$ is a *universal theory.* Then the CPS calculus with respect to $\mathcal{T}_0'$ is sound and refutationally complete for every $\mathcal{T}_0'$ that is obtained by adding free constants to $\mathcal{T}_0$.[20]

Theorem 54. *Let $\mathcal{T}_0 \subseteq \mathcal{T}_1 = \mathcal{T}_0 \cup N$ be a shallow theory extension with a set N of shallow clauses. Let G be a set of ground Σ_1-clauses, and let G_F be the flattened form of G. Then $\mathcal{T}_1 \cup G$ is unsatisfiable (in the total algebra semantics) if and only if the empty clause can be derived from* $\mathrm{abs}(N \cup G_F)$ *by constraint partial superposition with respect to $\mathcal{T}_0'$, where $\mathcal{T}_0'$ is obtained from $\mathcal{T}_0$ by adding the new total constants Ω_c to the base signature.*

Proof. Assume first that the empty clause can be derived from $\mathrm{abs}(N \cup G_F)$ by constraint partial superposition with respect to $\mathcal{T}_0'$. Then, by Thm. 47, $\mathrm{abs}(N \cup G_F)$ has no partial model which is a model of $\mathcal{T}_0'$, so $\mathrm{abs}(N \cup G_F)$ has no total model which is a model of $\mathcal{T}_0'$. As abstraction and flattening preserve [un-]satisfiability with respect to the total algebra semantics, it follows that $\mathcal{T}_1 \cup G_F$ is unsatisfiable with respect to total Σ_1'-algebras (where Σ_1' is obtained by adding the constants in Ω_c to Σ_1), hence $\mathcal{T}_1 \cup G$ is unsatisfiable with respect to total Σ_1'-algebras.

Assume now that the empty clause cannot be derived from $\mathrm{abs}(N \cup G_F)$ by constraint partial superposition with respect to $\mathcal{T}_0'$ (as $\mathcal{T}_0$ is a universal first-order theory, $\mathcal{T}_0'$ is also a universal first-order theory)[21]. By Thm. 47, $\mathrm{abs}(N \cup G_F)$ has a partial model A, such that $A|_{\Sigma_0'}$ is a total model of $\mathcal{T}_0'$. As abstraction preserves [un-]satisfiability with respect to partial algebras, A is also a partial model of $N \cup G_F$. Let B be the extension of A to a total Σ_1'-algebra obtained as explained in Thm. 51. By Thm. 51, B is a total model

[20] The results can be extended to more general base theories, but, for the sake of simplicity these extensions are not discussed here.

[21] A similar argument can be used for compact non-first-order base theories $\mathcal{T}_0$, under the additional assumption that $\mathcal{T}_0'$ is compact as well, and constraint partial superposition with respect to $\mathcal{T}_0'$ is complete.

of $\mathcal{T}_1'$ (where $\mathcal{T}_1'$ is obtained by adding the constants in Ω_c to $\mathcal{T}_1$). Note that, due to the form of the clauses in G_F, every clause that contains a partial function symbol f is a ground unit clause of the form $f(t_1, \dots, t_n) \approx t$, where the terms $t_1, \dots, t_n, t$ are totally defined in A. As A is a partial model of G_F, it follows that $f(t_1, \dots, t_n)$ is defined in A, so B is also a model of the unit clause $f(t_1, \dots, t_n) \approx t$. All the other clauses have all terms defined in A, thus hold also in B. Thus, B is a model of G_F, hence $\mathcal{T}_1 \cup G_F$ (and, therefore, also $\mathcal{T}_1 \cup G$) is satisfiable. $\square$

Theorem 55. *Suppose that $\mathcal{T}_0 \subseteq \mathcal{T}_1$ is a shallow theory extension. Let N be the set of (shallow) clauses in $\mathcal{T}_1 \setminus \mathcal{T}_0$ and let C be a Σ_1-clause with free variables $x_1, \dots, x_n$. Then $\mathcal{T}_1 \models \forall x_1 \dots \forall x_n\, C$ if and only if the empty clause can be derived from* $\mathrm{abs}(N \cup G_F)$ *by constraint partial superposition with respect to $\mathcal{T}_0'$ (that is, $\mathcal{T}_0$ plus the new constants in G_F), where G_F is the set of ground unit clauses obtained from $\exists x_1 \dots \exists x_n\, \neg C$ by skolemization followed by flattening.*

Proof. With the notation above it is obvious that the following statements are equivalent:

(a) $\mathcal{T}_1 \models \forall x_1 \dots \forall x_n\, C$ (in the total algebra semantics)
(b) $\mathcal{T}_1 \cup \exists x_1 \dots \exists x_n\, \neg C$ is unsatisfiable (in the total algebra semantics)
(c) $\mathcal{T}_0 \cup N \cup G_F$ has no total model.

By Thm. 54, $\mathcal{T}_0 \cup N \cup G_F$ has no total model if and only if the empty clause can be derived from $\mathrm{abs}(N \cup G_F)$ by constraint partial superposition. $\square$

Extensions of a base theory $\mathcal{T}_0$ with free function symbols are shallow extensions of $\mathcal{T}_0$. Therefore, a simple application of Thm. 54 is to the case where we want to prove unsatisfiability of sets of ground clauses over an extension of a theory with free function symbols: flattening the clauses followed by applying constraint partial superposition is a sound and refutationally complete (w. r. t. total algebra semantics) and modular method for this problem.

Corollary 56. *Let $\mathcal{T}_1$ be an extension of $\mathcal{T}_0$ by a set Ω_F of free function symbols. Then flattening and abstraction of the clauses followed by applying the CPS calculus (in which all functions in Ω_F are considered as partial) is a sound and refutationally complete method for testing the satisfiability in $\mathcal{T}_1$ of sets of ground clauses.*

Cor. 56 allows us to give a decision procedure for the universal fragment of an extension of a first-order theory $\mathcal{T}_0$ with free function symbols, under the assumption that the universal fragment of $\mathcal{T}_0$ is decidable.

Theorem 57. *Assume that $\mathcal{T}_0$ has a decidable universal (clause) theory. Then the universal theory of any extension $\mathcal{T}_1$ of $\mathcal{T}_0$ by a set Ω_F of free function symbols is also decidable.*

Proof. Let C be a clause with free variables $x_1, \dots, x_n$ in the signature $\Sigma_1 = (S_0, \Omega_0^{\mathrm{T}}, \Omega_F)$. Let G_F be the set of (ground unit) clauses obtained from skolemization followed by flattening from $\exists x_1 \dots \exists x_n \neg C$. Let $\mathcal{T}_0'$ (resp. $\mathcal{T}_1'$) be the theory $\mathcal{T}_0$ (resp. $\mathcal{T}_1$) plus the newly introduced constants, in signature Σ_0' (resp. Σ_1') (i.e. the signature Σ_i – where $i = 0$ or 1 resp. – plus the newly introduced constants).

By Thm. 54 (and Cor. 56), $\mathcal{T}_1 \models \forall x_1 \dots \forall x_n\, C$ if and only if the saturation of $\mathrm{abs}(G_F)$ under constraint partial superposition with respect to $\mathcal{T}_0'$ does not contain the empty clause. (Note again that the functions in Ω_F are declared as partial.)

One can see that G_F is the union of two sets G_0 and G_1 where G_0 consists only of (ground unit) Σ_0'-clauses, and G_1 only of Σ_1'-clauses of the form $f(t_1, \dots, t_n) \approx t$, where $f \in \Omega_F$ and $t_1, \dots, t_n, t$ are ground terms consisting only of total symbols. We analyze all possible constraint partial superposition inferences between clauses in $\mathrm{abs}(G_F)$.

Since all clauses which contain function symbols in Ω_F are ground flat positive unit clauses in G_F, the clauses in $\mathrm{abs}(G_1)$ are all Horn. Therefore, no inferences by *partial top-superposition*, *merging paramodulation*, and *factoring* between clauses in $\mathrm{abs}(G_1)$ are possible. Thus, the only inferences in the CPS calculus involving clauses which contain function symbols in Ω_F are *superposition* inferences between two clauses in $\mathrm{abs}(G_1)$. These can only be of the form:

$$\frac{x \not\approx s \vee \bigvee_{i=1}^{n} x_i \not\approx s_i \vee f(x_1, \dots, x_n) \approx x \qquad y \not\approx t \vee \bigvee_{i=1}^{n} y_i \not\approx t_i \vee f(y_1, \dots, y_n) \approx y}{x \not\approx s \vee \bigvee_{i=1}^{n} x_i \not\approx s_i \vee y \not\approx t \vee \bigvee_{i=1}^{n} x_i \not\approx t_i \vee x \approx y}$$

The resulting clause is always a Σ_0' clause. Therefore, testing the satisfiability of $\mathcal{T}_0 \cup G_F$ can be done in the following steps:

(1) Saturate G_1 under *superposition* (this can be done in quadratic time in $|G_1|$; a set N_1 of Σ_0'-clauses is generated, where N_1 contains, up to renaming of variables, at most $|G_1|^2$ Σ_0'-clauses).
(2) If $\perp$ is not generated during step (1), test the satisfiability of $\mathcal{T}_0' \cup G_0 \cup N_1$ by *constraint refutation.*

Note that every clause in N_1 is of the form

$$x \not\approx s \vee \bigvee_{i=1}^{n} x_i \not\approx s_i \vee y \not\approx t \vee \bigvee_{i=1}^{n} x_i \not\approx t_i \vee x \approx y$$

and therefore is equivalent to the ground clause $\bigvee_{i=1}^{n} s_i \not\approx t_i \vee s \approx t$. Thus, N_1 is equivalent with a set N_1^g of ground Σ_0'-clauses which contains at most

$|G_1|^2$ clauses, each of length at most $n+1$, where n is the maximal arity of the function symbols in Ω_F.

Since we assumed that the universal theory of $\mathcal{T}_0$ is decidable, the universal theory of $\mathcal{T}_0'$ is also decidable and has the same complexity. Indeed, it is easy to see that the following statements are equivalent:

(a) $\mathcal{T}_0' \models \forall y_1 \dots \forall y_k \, D(c_1, \dots, c_m, y_1, \dots, y_k)$;
(b) $\mathcal{T}_0' \cup \exists y_1 \dots \exists y_k \, \neg D(c_1, \dots, c_m, y_1, \dots, y_k)$ is unsatisfiable (with respect to $\Sigma_0 \cup \Omega_c$-algebras);
(c) $\mathcal{T}_0' \cup \neg D(c_1, \dots, c_m, d_1, \dots, d_k)$ is unsatisfiable (with respect to $\Sigma_0 \cup \Omega_c \cup \Omega_{\mathsf{sk}}$-algebras), where $\Omega_{\mathsf{sk}} = \{d_1, \dots, d_k\}$ is a new set of Skolem constants which replace the existentially quantified variables $\{y_1, \dots, y_k\}$;
(d) $\mathcal{T}_0 \cup \exists x_1 \dots \exists x_n \, \exists y_1 \dots \exists y_k \, \neg D(x_1, \dots, x_m, y_1, \dots, y_k)$ is unsatisfiable (with respect to Σ_0-algebras);
(e) $\mathcal{T}_0 \models \forall x_1 \dots \forall x_n \, \forall y_1 \dots \forall y_k \, D(x_1, \dots, x_m, y_1, \dots, y_k)$.

Therefore, testing satisfiability of ground clauses w. r. t. $\mathcal{T}_0'$ is also decidable. Assume that there exists a function g such that for every input set G of ground unit clauses satisfiability of $\mathcal{T}_0 \cup G$ can be checked in time at most $g(n)$, where n is the size of G, i. e., the total number of symbols in G. Then for every set G of ground unit clauses of size n satisfiability of $\mathcal{T}_0' \cup G$ can be checked in time at most $g(n)$. Since the size of the input for the decision procedure for $\mathcal{T}_0'$ is quadratic in the size of the input for the original problem, the complexity of deciding the clause validity in $\mathcal{T}_1$ has as upper bound $g(k \cdot n^2)$, where n is the size of the input, and k is a constant natural number. □

This provides an alternative proof of a result established (for arbitrary theories) also in (Ganzinger [13], [22] Tinelli and Zarba [28]).

5.2 Local extensions of a base theory

A more general, but related, case is that of local extensions of a base theory (Sofronie-Stokkermans [25]). The definitions we present here are somewhat more restricted than those in [25], as they refer only to extensions with flat sets of clauses and flat goals.

Definition 58. A term is called *variable-flat* if all its proper subterms are variables. A *variable-flat theory extension* is an extension $\mathcal{T}_0 \subseteq \mathcal{T}_1$ by means of a set N of clauses, i. e., $\mathcal{T}_1 = \mathcal{T}_0 \cup N$ where N consists of clauses for which all subterms starting with a partial function are variable-flat.

[22] The result as such is not explicitly stated in [13], but is an immediate consequence of the results presented there.

Let N be a set of clauses, and let G be a set of ground clauses. We denote by $N[G]$ the set of the most general instances of clauses in N in which each subterm starting with a partial function is a ground subterm occurring in G or in N [23]. If N and G are finite, then the set $N[G]$ is finite and can be effectively computed from G.

Definition 59. Suppose $\mathcal{T}_0 \subseteq \mathcal{T}_1$ is a flat theory extension by means of a flat set N of clauses, i. e., $\mathcal{T}_1 = \mathcal{T}_0 \cup N$. We say that the extension $\mathcal{T}_0 \subseteq \mathcal{T}_1$ is *local* if, for every set Ω_c of constants and for every set G of flat ground Σ_1'-clauses (where Σ_1' equals Σ_1 plus the constants in Ω_c), the following are equivalent:

(i) $\mathcal{T}_1 \cup G$ is unsatisfiable in the total algebra semantics
(ii) $\mathcal{T}_0 \cup N[G] \cup G$ is unsatisfiable in the partial algebra semantics, where function symbols in $\Sigma_1 \backslash \Sigma_0$ are declared as partial, and all constants in Ω_c are declared as total.

This definition is related to the notion of local equational theory introduced in [12] and of locality in general [16,22].

Example 60 (Sofronie-Stokkermans [25]). The following theory extensions are local:

(1) **Extensions with free functions:** Any extension of a theory $\mathcal{T}_0$ with a set of free function symbols is local.
(2) **Extensions with selector functions:** Let $\mathcal{T}_0$ be a theory with signature $\Sigma_0 = (\Omega_0^T, \emptyset)$, let $c \in \Omega_0^T$ with arity n, and let $\Sigma_1 = (\Omega_1^T, \Omega_2^P)$, where $\Omega_1^T = \Omega_0^T$ and $\Omega_1^P = \{s_1, \ldots, s_n\}$ consists of n unary function symbols. Let $\mathcal{T}_1 = \mathcal{T}_0 \cup \mathsf{Sel}$ (a theory with signature Σ_1) be the extension of $\mathcal{T}_0$ with the set Sel of clauses below. If $\mathcal{T}_0$ satisfies the (universally quantified) formula $\mathsf{Inj(c)}$ (i. e. c is injective in $\mathcal{T}_0$) then the extension $\mathcal{T}_0 \subseteq \mathcal{T}_1$ is local.

$$\begin{array}{ll}
(\mathsf{Sel}) & s_1(c(x_1, \ldots, x_n)) \approx x_1 \\
 & \ldots \\
 & s_n(c(x_1, \ldots, x_n)) \approx x_n \\
(\mathsf{Inj(c)}) & c(x_1, \ldots, x_n) \approx c(y_1, \ldots, y_n) \rightarrow (\bigwedge_{i=1}^{n} x_i \approx y_i)
\end{array}$$

(3) **Extensions with monotone functions:** Let $\mathcal{T}_0$ be one of the following theories: (1) $\mathcal{P}$ (posets), (2) $\mathcal{T}$ (totally ordered sets), (3) $\mathcal{DO}$ (dense totally ordered sets), (4) $\mathcal{S}$ (semilattices), (5) $\mathcal{L}$ (lattices), (6) $\mathcal{DL}$ (distributive lattices), (7) $\mathcal{B}$ (Boolean algebras), (8) $\mathbb{R}$ (theory of reals), where we regard the predicate symbol $\leq$ as a total binary function with output

[23] Formally, $N[G] = \{\, C\sigma \mid C \in N$, and if the term $f(t_1, \ldots, t_n)$ in C starts with a partial function f then $f(t_1, \ldots, t_n)\sigma$ is a ground term in N or G and all variables not occurring below partial functions are unchanged by $\sigma\,\}$.

sort bool. Let Mon_f be the monotonicity axiom:

$$(\mathsf{Mon}_f) \quad \bigwedge_{i=1}^{n} x_i \leq y_i \rightarrow f(x_1, \dots, x_n) \leq f(y_1, \dots, y_n).$$

The extension $\mathcal{T}_0 \subseteq \mathcal{T}_0 \cup \mathsf{Mon}_f$ is local.

Shallow extensions satisfy a weaker notion of locality (namely stable locality) which is discussed in (Sofronie-Stokkermans [25]).

We now show that for a *variable-flat local theory extension* $\mathcal{T}_0 \subseteq \mathcal{T}_1 = \mathcal{T}_0 \cup N$ abstraction followed by constraint superposition is a refutationally complete method (w. r. t. total semantics) when applied to $\mathcal{T}_0 \cup N[G_F] \cup G_F$, and also to $\mathcal{T}_1 \cup G_F$, where G is a ground goal and G_F the flattened form of G. Thus, we have the choice between computing the instances in $N[G_F]$ or avoiding to do so – as this may be too expensive in many cases.

Theorem 61. *Let $\mathcal{T}_0$ be a first-order Σ_0-theory, and let $\mathcal{T}_1 = \mathcal{T}_0 \cup N$ be a variable-flat local extension of $\mathcal{T}_0$. Let G be a set of ground clauses, and G_F be the set of flat ground clauses obtained from flattening G. Then the following are equivalent:*

(1) $\mathcal{T}_1 \cup G$ is unsatisfiable.
(2) The empty clause can be obtained from $\mathrm{abs}(\mathcal{T}_0 \cup N[G_F] \cup G_F)$ *by constraint superposition (in which all functions in $\Sigma_1 \backslash \Sigma_0$ are supposed to be partial) w. r. t. $\mathcal{T}_0'$ (i. e. the base theory $\mathcal{T}_0$ plus the newly introduced constants).*
(3) The empty clause can be obtained from $\mathrm{abs}(\mathcal{T}_1 \cup G_F)$ *by constraint superposition (in which all functions in $\Sigma_1 \backslash \Sigma_0$ are supposed to be partial) w. r. t. $\mathcal{T}_0'$.*

Proof. Consider the following statements:

(a) $\mathcal{T}_1 \cup G$ has a total algebra model;
(b) $\mathcal{T}_1 \cup G_F = \mathcal{T}_0 \cup N \cup G_F$ has a total algebra model;
(c) $\mathcal{T}_0 \cup N[G_F] \cup G_F$ has a partial algebra model;
(d) $\mathcal{T}_0 \cup N \cup G_F$ has a partial algebra model.

As flattening preserves [un-]satisfiability w. r. t. the total algebra semantics, (a) and (b) are equivalent. Clearly, (b) implies (c): a total model of $\mathcal{T}_0 \cup N \cup G_F$ satisfies all instances of N, hence in particular $N[G_F]$. As $\mathcal{T}_1$ is a local extension of $\mathcal{T}_0$, (c) implies (b). As every total model is a partial model, (b) implies (d). We prove that (d) implies (c).

Let A be a partial model of $\mathcal{T}_0 \cup N \cup G_F$. We show that A is also a model for $N[G_F]$. Let $\overline{D} \in N[G_F]$. Then there exists a clause $D \in N$ and a substitution $\sigma : X \rightarrow T_{\Sigma_1}(X)$ with $D\sigma = \overline{D}$ and every term in $D\sigma$ which starts with a partial function symbol is a ground term in G_F. As G_F is flat, all clauses in G_F which contain a partial function symbol are unit ground clauses of the form $f(c_1, \dots, c_n) \approx c$. As N is variable-flat, partial functions in N have as arguments only variables.[24]

As A is a partial model of $N \cup G_F$ it follows that all ground subterms of N or G_F which start with a partial function symbol are totally defined in A. Thus, for every assignment β into A, all subterms of $\overline{D}$ are defined in (A, β). Let β be an arbitrary assignment into A, and let $\gamma : X \to A$ defined by $\gamma(x) := (A, \beta)(\sigma(x))$. Then $(A, \beta)(\sigma(t))$ (and hence also $\gamma(t)$) is defined in A for every subterm t of D. As A is also a model for N, and all subterms of D are defined, $(A, \beta)(D\sigma) = (A, \gamma)(D) = 1$. Thus, A is a model of $\overline{D}$. Hence, if A is a model of N then A is a model of $N[G_F]$.

This shows that (a) and (d) are equivalent. By Theorem 47, $\mathcal{T}_0 \cup N[G_F] \cup G_F$ has a partial algebra model if and only if the empty clause cannot be obtained from $\text{abs}(N[G_F] \cup G_F)$ by constraint superposition with respect to $\mathcal{T}_0'$, and $\mathcal{T}_0 \cup N \cup G_F$ has a partial algebra model if and only if the empty clause cannot be obtained from $\text{abs}(N \cup G_F)$ by constraint superposition with respect to $\mathcal{T}_0'$. This completes the proof of the theorem. $\square$

Corollary 62. *Let $\mathcal{T}_0$ be a first-order Σ_0-theory, and let $\mathcal{T}_1 = \mathcal{T}_0 \cup N$ be a local extension of $\mathcal{T}_0$. Abstracting and then applying the CPS calculus (in which all functions in $\Sigma_1 \backslash \Sigma_0$ are supposed to be partial) is a sound and complete method for testing the validity of universally quantified formulae in $\mathcal{T}_1$.*

Proof. Let C be a clause with free variables $x_1, \dots, x_n$ in the signature Σ_1. $\mathcal{T}_1 \models \forall x_1 \dots \forall x_n C$ if and only if $\mathcal{T}_1 \cup \exists x_1 \dots \exists x_n \neg C \models \perp$. Let G be the set of (unit, ground) clauses obtained from Skolemization (possibly followed by flattening) from $\exists x_1 \dots \exists x_n \neg C$. By Theorem 61, $\mathcal{T}_1 \cup G$ is unsatisfiable if and only if the empty clause can be obtained from $\text{abs}(N[G] \cup G)$ (resp. from $\text{abs}(N \cup G)$) by constraint superposition with respect to $\mathcal{T}_0'$ (the base theory $\mathcal{T}_0$ plus the newly introduced constants). $\square$

The results in this section show that for shallow and local extensions of a base theory flattening and abstraction of the clauses followed by applying the CPS calculus (in which all functions in Ω_F are supposed to be partial) is not only a sound, but also a refutationally complete method for testing satisfiability in $\mathcal{T}_1$ of sets of ground clauses, resp. for testing the validity of the universal theory.

[24] Therefore σ must assign constants to all variables occurring below a partial function symbol and should leave unchanged all variables which do not occur below a partial function.

6 Related Work

In this section related work is summarized and compared with the results presented in the paper.

Validity of identities in partial algebras.

Evans validity is often related to properties of embeddability of partial algebras into total algebras [11,7,17]. This connection allows us to replace equational reasoning for total functions with reasoning about partial functions, or with relational reasoning. Evans validity was also used in (Ganzinger [12]) for establishing relationships between semantic and proof-theoretic approaches to polynomial time decidability for uniform word problems for quasi-varieties, in particular connections between embeddability and locality of equational theories.

Besides Evans validity [11,7,17] there are many other possibilities for defining validity of identities in partial algebras, from which we mention only a few (for further details we refer to Burmeister [6]):

- existential validity: $(A, \beta) \models t \stackrel{e}{\approx} t'$ if and only if $(A, \beta)(t)$ and $(A, \beta)(t')$ are both defined and equal;
- strong validity: $(A, \beta) \models t \stackrel{s}{\approx} t'$ if and only if either both $(A, \beta)(t)$ and $(A, \beta)(t')$ are undefined or both are defined and equal;
- weak validity: $(A, \beta) \models t \stackrel{w}{\approx} t'$ if and only if $(A, \beta)(t)$ or $(A, \beta)(t')$ are undefined or both are defined and equal.

Both existential and strong validity are in some sense less constructive than Evans validity: If b is a total constant and f and g are partial functions, then both $\mathsf{f}(\mathsf{g}(\mathsf{b})) \stackrel{e}{\approx} \mathsf{b}$ and $\mathsf{f}(\mathsf{g}(\mathsf{b})) \stackrel{s}{\approx} \mathsf{b}$ imply that $\mathsf{f}(\mathsf{g}(\mathsf{b}))$ is defined. Under the assumption that functions are strict, this implies that $\mathsf{g}(\mathsf{b})$ is defined, but there is no way to "compute" $\mathsf{g}(\mathsf{b})$, i. e., to express it in terms of total functions. [25] It is therefore not clear how one could modify our calculi in order to make them usable for existential or strong validity.

Using our results for weak validity is unproblematic due to the following encoding trick: Let id be a (total or partial) function satisfying the axiom $id(x) \approx x$. Then $(A, \beta) \models t \stackrel{w}{\approx} t'$ if and only if $(A, \beta) \models id(t) \approx id(t')$ in Evans validity. To use weak validity instead of Evans validity, even on a per clause or

[25] Extending the signature with additional total function symbols to give explicit definitions for all defined subterms fixes this problem. In the modular or hierarchic case, adding total functions with the required codomain may be impossible, though.

per literal basis, it is therefore sufficient to replace positive literals $t \approx t'$ by $id(t) \approx id(t')$. Negative literals are not changed.

Resolution calculi for partial functions and partial congruences.

An alternative way to dealing with undefinedness, which goes back to Kleene [21], is to use many-valued logic, with an additional truth value for "undefined". Kleene's logic has been used by various authors for giving logical systems for partial functions and for reasoning about partial functions in a many-valued framework. A resolution calculus for partial functions, where undefinedness is formalized using Kleene's strong three valued logic, was proposed by Kerber and Kohlhase in [20]. Although we also use a three-valued logic for modeling undefinedness, where the negation is similar to Kleene's strong negation, the notion of validity used in [20] is different from the one we use, as no distinction is made between undefinedness and irrelevance of a term. [26] The calculus presented in this paper is different from the one in [20] on the one hand because of the different notion of validity mentioned and on the other hand because refinements of resolution such as paramodulation or superposition are not considered in [20].

Bachmair and Ganzinger [4] give a version of ordered chaining for partial equivalence and congruence axioms. This calculus is devised for *strong* or *existential validity*; consequently, *equality resolution* is replaced with a rule which encodes partial reflexivity. In particular, in [4] one can make statements about definedness of certain terms (more precisely, a term is defined if $t \approx t$ is derivable in the calculus). In contrast, Evans' validity does not allow one to define totality of a partial function or of a term. Therefore, the calculus we describe in this paper is different from the one in [4].

Superposition-based reasoning

The CPS calculus resembles a calculus presented by Bachmair, Ganzinger, and Waldmann [5], where a base theory is extended by *total* functions, but where sufficient completeness of the extension is necessary for the refutational completeness of the calculus. Due to the different logics used, the calculi are not fully comparable, though. In particular, the CPS calculus does not subsume the hierarchic superposition calculus of [5] for sufficiently complete ex-

[26] In [20], an atomic formula $P(t_1, \ldots, t_n)$ has truth value "undefined" if at least one of $t_1, \ldots, t_n$ is undefined. This is in fact a notion of weak validity (which, as pointed out before, can be modelled in our framework by using an additional unary function symbol).

tensions.[27]

Superposition was used for reasoning in combinations of theories over signatures sharing only constants by Hillenbrand [18]. In (Armando, Rusinowitch, and Ranise [2]) superposition with selection is tuned to specific theories (such as lists and arrays, possibly with extensionality) and proved to yield decision procedures with optimal complexity. These results are used for proving that modular reasoning is possible in a very special case of combinations of theories (lists and arrays). The problem remains that one usually wants to consider more complex theories, e. g., extensions of a base theory of elements with new sorts and additional functions for describing data structures (such as lists or arrays) over the theory of elements and operations on these data structures. One possibility for dealing with this situation, which goes back to Nelson and Oppen, is to devise methods for reasoning in a combination of a suitable theory of lists (resp. arrays) with a theory of elements. This works if the theory of elements is stably infinite. Similar methods are applied for reasoning in certain combinations of theories (Armando, Bonacina, Ranise, and Schulz [1]).

Our approach is different. We show that if the extensions only introduce additional partial functions, a superposition calculus for partial functions becomes a complete and modular proof system where inferences are pure. We also analyze situations in which similar modularity results can be obtained for combining extensions with *total* instead of partial functions. In this framework, stable infiniteness of the theories is not needed for refutational completeness (for instance stable infiniteness of the base theory of elements that is extended to a list theory).

Modular theorem proving in combinations of theories.

In Nelson/Oppen-style combinations of stably infinite theories $\mathcal{T}_1$ and $\mathcal{T}_2$ over signatures Σ_1 and Σ_2 which are disjoint or share only constants, inferences are always pure. Ghilardi [15] has recently extended the completeness results for modular inference systems for combinations of theories over non-disjoint signatures. Thm. 29, one of the main results of our paper, also provides a modular way of combining extensions $\mathcal{T}_1$ and $\mathcal{T}_2$ of a base theory $\mathcal{T}_0$. The main difference between Ghilardi's approach and our work is that in (Ghilardi [15]) the component theories need to satisfy a rather strong compatibility condition with respect to the shared theory. On the other hand, our calculi are only complete with respect to the partial function semantics. We have shown, however, that for shallow or local extensions of base theories partial models can always be made total. Ghilardi's compatibility conditions ensure, in addition,

[27] The main obstacle is the fact that sufficient completeness w. r. t.. simple instances as defined in [5] (using total semantics) is not equivalent to its counterpart sufficient completeness w. r. t.. total instances (using partial semantics).

that the Craig interpolants consist of positive ground clauses whereas in the modular partial superposition calculus described in this paper clauses with variables need to be exchanged between the theory modules.

For Thm. 29 to be applicable, the theories $\mathcal{T}_1$ and $\mathcal{T}_2$ (regarded as theories with partial functions in Σ_1, Σ_2) most have the same total function symbols. A similar situation was analyzed by Tinelli [26], who gives a method for co-operation of background reasoners for universal theories which have *the same function symbols*. However, we have shown that there are interesting problem classes where partial models can always be totalized. Therefore, in these cases the condition that the theories $\mathcal{T}_1$ and $\mathcal{T}_2$ have the same total function symbols can be relaxed. The superposition calculus for partial functions developed in this paper also allows us to efficiently compute the (universal) Craig interpolant even in this more general case.

7 Conclusions

In this paper we have presented a partial superposition calculus for the combination of first-order theories involving both total and partial (many-sorted) functions. We have shown that the calculus is modular provided that functions that are not in the intersection of the component signatures are declared as partial.

We have also considered a constraint superposition calculus for hierarchical theories and proved that it has a related modularity property. We have shown that constraint partial superposition is complete whenever every function in the extension whose codomain is a base sort is declared as partial; a function whose codomain is an extension sort can either be declared as total or partial.

An important application of this result is to approximate refutational theorem proving in extensions of base theories for which refutationally complete black box theorem provers exist. If constraint partial superposition finds a contradiction for a set of clauses in the extended signature, the set is unsatisfiable in particular also with respect to total algebras. Therefore, in this way we obtain a sound approximation of refutational theorem proving in extensions of first-order theories. We have shown that if every partial algebra can be "completed" to a total algebra then this approximation is complete. This is the case for shallow extensions of a base theory, e. g., extensions of a base theory with functions defined by tail-recursion. Another case (and a generalization) are local extensions of a base theory.

We expect to be able to use the calculi developed in this paper for obtaining efficient algorithms for modular reasoning in combinations of many-sorted complex theories. Dealing efficiently with partial functions can be a goal in itself, but the results on local theory extensions which we consider indicate that that the range of expected results is wider.

References

[1] Alessandro Armando, Maria Paola Bonacina, Silvio Ranise, and Stephan Schulz. On a rewriting approach to satisfiability procedures: extension, combination of theories and an experimental appraisal. In *Proceedings of the 5th International Workshop Frontiers of Combining Systems (FroCos'05)*, 2005, LNAI. Springer Verlag. To appear.

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

[3] Leo Bachmair and Harald Ganzinger. Rewrite-based equational theorem proving with selection and simplification. *Journal of Logic and Computation*, 4(3):217–247, 1994.

[4] Leo Bachmair and Harald Ganzinger. Ordered chaining calculi for first-order theories of transitive relations. *Journal of the ACM*, 45(6), 1998.

[5] Leo Bachmair, Harald Ganzinger, and Uwe Waldmann. Refutational theorem proving for hierarchic first-order theories. *Applicable Algebra in Engineering, Communication and Computing*, 5(3/4):193–212, 1994.

[6] Peter Burmeister. *A Model Theoretic Oriented Approach to Partial Algebras: Introduction to Theory and Application of Partial Algebras, Part I*, vol. 31 of *Mathematical Research*. Akademie-Verlag, Berlin, 1986.

[7] Stanley Burris. Polynomial time uniform word problems. *Mathematical Logic Quarterly*, 41:173–182, 1995.

[8] Hubert Comon, Marianne Haberstrau, and Jean-Pierre Jouannaud. Decidable problems in shallow equational theories (extended abstract). In *Seventh Annual IEEE Symposium on Logic in Computer Science*, Santa Cruz, California, USA, 1992, pp. 255–265. IEEE Computer Society Press, Los Alamitos, CA, USA.

[9] Vincent J. Digricoli and Malcolm C. Harrison. Equality-based binary resolution. *Journal of the ACM*, 33(2):253–289, 1986.

[10] Trevor Evans. The word problem for abstract algebras. *Journal of the London Mathematical Society*, 26:64–71, 1951.

[11] Trevor Evans. Embeddability and the word problem. *Journal of the London Mathematical Society*, 28:76–80, 1953.

[12] Harald Ganzinger. Relating semantic and proof-theoretic concepts for polynomial time decidability of uniform word problems. In *Sixteenth Annual IEEE Symposium on Logic in Computer Science*, Boston, Massachusetts, USA, June 2001, pp. 81–90. IEEE Computer Society, Los Alamitos, CA, USA.

[13] Harald Ganzinger. Shostak light. In Andrei Voronkov, ed., *Automated Deduction – CADE-18*, Copenhagen, Denmark, 2002, LNCS 2392, pp. 332–346. Springer-Verlag.

[14] Harald Ganzinger, Viorica Sofronie-Stokkermans, and Uwe Waldmann. Modular proof systems for partial functions with weak equality. In David Basin and Michaël Rusinowitch, eds., *Automated Reasoning: Second International Joint Conference, IJCAR 2004*, Cork, Ireland, 2004, LNAI 3097, pp. 168–182. Springer-Verlag. Corrected version at `http://www.mpi-sb.mpg.de/~uwe/paper/PartialFun-bibl.html`.

[15] Silvio Ghilardi. Quantifier elimination and provers integration. *Electronic Notes in Theoretical Computer Science*, 86(1), 2003.

[16] Robert Givan and David McAllester. New results on local inference relations. In *Principles of Knowledge Representation and reasoning: Proceedings of the Third International Conference (KR'92)*, 1992, pp. 403–412. Morgan Kaufmann Press.

[17] George Grätzer. *Universal algebra.* Springer-Verlag, 2nd edition, 1968.

[18] Thomas Hillenbrand. A superposition view on Nelson-Oppen. In Ulrike Sattler, ed., *Contributions to the Doctoral Programme of the Second International Joint Conference on Automated Reasoning*, 2004, vol. 106 of *CEUR Workshop Proceedings*, pp. 16–20.

[19] Florent Jacquemard, Christoph Meyer, and Christoph Weidenbach. Unification in extensions of shallow equational theories. In Tobias Nipkow, ed., *Rewriting Techniques and Applications, 9th International Conference, RTA-98*, 1998, LNCS 1379, pp. 76–90. Springer-Verlag.

[20] Manfred Kerber and Michael Kohlhase. A mechanization of strong Kleene logic for partial functions. In Alan Bundy, ed., *Twelfth International Conference on Automated Deduction*, Nancy, France, 1994, LNAI 814, pp. 371–385. Springer-Verlag.

[21] Stephen C. Kleene. *Introduction to Metamathematics.* D. Van Nostrand Company, Inc., Princeton, New Jersey, 1952.

[22] David McAllester. Automatic recognition of tractability in inference relations. *Journal of the Association for Computing Machinery*, 40(2):284–303, 1993.

[23] Greg Nelson and Derek C. Oppen. Simplification by cooperating decision procedures. *ACM Transactions on Programming Languages and Systems*, 1(2):245–257, 1979.

[24] Robert Nieuwenhuis. Basic paramodulation and decidable theories. In *Eleventh Annual IEEE Symposium on Logic in Computer Science*, New Brunswick, NJ, USA, 1996.

[25] Viorica Sofronie-Stokkermans. Hierarchic reasoning in local theory extensions. In Robert Nieuwenhuis, ed., *20th International Conference on Automated Deduction – CADE-20*, Tallinn, Estonia, 2005, LNAI 3632, pp. 219–234. Springer-Verlag.

[26] Cesare Tinelli. Cooperation of background reasoners in theory reasoning by residue sharing. *Journal of Automated Reasoning*, 30(1):1–31, 2003.

[27] Cesare Tinelli and Mehdi Harandi. A new correctness proof of the Nelson-Oppen combination procedure. In Franz Baader and Klaus U. Schulz, eds., *Frontiers of Combining Systems, First International Workshop*, Munich, Germany, 1996, Applied Logic Series, Vol. 3, pp. 103–119. Kluwer Academic Publishers.

[28] Cesare Tinelli and Calogero Zarba. Combining non-stably infinite theories. *Electronic Notes in Theoretical Computer Science*, 86(1), 2003.