Theory Instantiation

Harald Ganzinger[1] and Konstantin Korovin[2*]

[1] MPI für Informatik
[2] The University of Manchester

Abstract. In this paper we present a method of integrating theory reasoning into the instantiation framework. This integration is done in the black-box style, which allows us to integrate different theories in a uniform way. We prove completeness of the resulting calculus, provided that the theory reasoner is answer-complete and complete for reasoning with ground clauses. One of the distinctive features of our approach is that it allows us to employ off-the-shelf satisfiability solvers for ground clauses modulo theories, as a part of general first-order reasoning. As an application of this approach, we show how it is possible to combine the instantiation calculus with other calculi, such as ordered resolution and paramodulation.

1 Introduction

Instantiation-based theorem proving has been studied intensively in recent years, see, e.g., [6, 7, 15, 17, 18, 23] among others. It has attractive features of combining efficient reasoning on ground formulas with first-order reasoning.

In this paper we develop a method for integrating theory reasoning into the instantiation-based framework, introduced in [13]. Approaches for integrating theory reasoning into a logical calculus usually fall into two major categories: black-box and glass-box approaches.

In the glass-box approach, theory reasoning is integrated via specialised inference rules, e.g., for the theory of equality, ordered paramodulation can be used. Usually, the resulting calculus is very efficient for a particular theory. However, for each theory one needs to devise specific rules which can make completeness arguments for the resulting calculus highly non-trivial. There is extensive literature on the integration of various theories into the resolution based framework. Much less is known about such integration into the instantiation framework beyond the integration of equality reasoning [8, 14, 18, 23].

In this paper we introduce theory instantiation which is closely related to theory resolution [24], and can be viewed as a black-box approach. Thus we assume only limited knowledge of the theory itself and the theory reasoner. This allows us to integrate theory reasoning in a uniform manner for different theories.

We follow the instantiation framework developed in our earlier papers [13, 14]. In our theorem proving process we interleave efficient satisfiability checking for

* Supported by the EPSRC grant GR/T08760/01.

ground clauses with appropriate instantiations witnessing unsatisfiability at the ground level. One of the distinctive features of our approach is that it allows us to employ off-the-shelf satisfiability solvers for ground clauses modulo theories, as a part of general first-order reasoning. Let us note that for many important theories such reasoners have received considerable attention and very efficient implementations are available (see, e.g., [5] and work on DPLL(T) [11, 21, 25]).

We prove completeness of the resulting calculus provided that the theory reasoner satisfies some general requirements. In particular, we require the theory reasoner to be answer-complete for reasoning with unit clauses, and to be complete for reasoning with ground clauses (formal definitions are given later in the paper). Our completeness proof is based on the model generation technique (see [3, 22]), which allows us to justify redundancy elimination based on a semantic notion of redundant clauses and redundant inferences. We also show that the instantiation process can be guided by (partial) information on models for ground clauses.

One of the applications of the presented approach is a method for combining various calculi with instantiation. It is reasonable to assume that some classes of formulas can be efficiently treated by instantiation, e.g., near propositional formulas, whereas other classes by resolution/paramodulation calculi. Therefore, combinations of various calculi is an important issue. In our approach we can divide the set of input clauses into two classes: the first class can be taken as theory clauses and we apply a specialised procedure to them; the second class are clauses treated with the instantiation calculus. In this case, the theory reasoner itself can be a logical calculus which satisfies the abstract requirements on the theory reasoner. We show that the requirements on the theory reasoner can be naturally satisfied by the ordered paramodulation calculus. Let us note that in this setting it is natural to use the black-box approach since the theory axiomatized by the theory part is generally not known in advance.

Our approach for integrating theory reasoning is closely related to theory resolution [9, 16, 24], (see also work on DPLL(T) [11, 25]). Here we consider full first-order reasoning in the instantiation-based framework. We are also concerned with issues of how to restrict instantiation and issues related to permutative theories.

2 Preliminaries

Let $\Sigma = \langle \mathcal{P}, \mathcal{F} \rangle$ be a first-order signature, where $\mathcal{P}$ is the set of predicate symbols and $\mathcal{F}$ is the set of function symbols. We assume that $\mathcal{P}$ contains the equality predicate $\simeq$ and $\mathcal{F}$ contains the constant $\perp$. The term algebra $\mathcal{T}(\mathcal{F})$, with the universe of all ground terms in $\mathcal{F}$ is defined as usual by assigning an interpretation of a function $f^{\mathcal{T}(\mathcal{F})}(t_1, \ldots, t_n)$ to $f(t_1^{\mathcal{T}(\mathcal{F})}, \ldots, t_n^{\mathcal{T}(\mathcal{F})})$.

A clause is a possibly empty multiset of literals denoting their disjunction and is usually written as $L_1 \vee \ldots \vee L_n$; a literal being either an atomic formula or the negation thereof. The logical constant false is denoted as $\square$. Variables are usually denoted by x, y, and z, whereas, unless indicated otherwise, letters

a, b and c denote constants. If L is a literal, $\overline{L}$ denotes the complement of L. Substitutions are defined as usual and will be denoted by letters ρ, σ, τ, and θ. We will also use $\bot$ to denote the substitution mapping all variables to the constant $\bot$. If S is a set of clauses, by $S\bot$ we denote all ground clauses obtained by applying $\bot$ to each clause in S. *Renamings* are injective substitutions which map variables to variables. Two clauses are *variants* of each other if one can be obtained from the other by applying a renaming.

As in our previous work on instantiation ([13, 14]), we consider a refined notion of instances of clauses called closures. A *closure* is a pair consisting of a clause C and a substitution σ written $C \cdot \sigma$. We work modulo renaming, that is, do not distinguish between closures $C \cdot \sigma$ and $D \cdot \tau$ for which C is a variant of D and $C\sigma$ is a variant of $D\tau$. A closure is called *ground* if it represents a ground clause. Let S be a set of clauses and C be a clause in S, then a ground closure $C \cdot \sigma$ is called a *ground instance* of S; we also say that the closure $C \cdot \sigma$ is a *representation (of the clause $C\sigma$) in S*. Truth values for closures are defined from the truth values of the clauses they represent.

We consider a universally axiomatized background theory T in the signature Σ [3] and assume that T implies the usual congruence axioms for equality. We are interested in proving unsatisfiability of sets of clauses modulo T, i.e., in proving that there is no model of T in which the considered clauses are true. Since the theory T is universal, from the Herbrand theorem it follows that a set of clauses is satisfiable in a model of T if and only if it is satisfiable in a Herbrand model of T. Therefore, we can restrict ourselves to Herbrand models. A Herbrand model of T, called *T-model*, is a model of T on the set of all ground terms where all functions are interpreted as in the term algebra $\mathcal{T}(\mathcal{F})$.

We say that a clause C follows from clauses $C_1, \ldots, C_n$ modulo T (or *T-follows*), denoted $C_1, \ldots, C_n \models_T C$, if for all T-models where $C_1, \ldots, C_n$ are true, C is also true. We say that a clause C is *T-satisfiable* if C is true in a T-model, likewise we say C is *T-unsatisfiable* if C is false in all T-models, also denoted as $C \models_T \Box$. For two ground clauses C, C' we write $C \leftrightarrow_T C'$ if $\models_T C \leftrightarrow C'$. For two ground terms t and s, we write $t \simeq_T s$ if $\models_T t \simeq s$, and $t \not\simeq_T s$ if $\not\models_T t \simeq s$. Likewise, we write for a pair of n-tuples of ground terms $\bar{t} \simeq_T \bar{s}$ if $t_i \simeq_T s_i$ for all $1 \leq i \leq n$ and write $\bar{t} \not\simeq_T \bar{s}$ if for some $1 \leq i \leq n$, $t_i \not\simeq_T s_i$. Let us note that $\simeq_T$ is the least congruence on the term algebra $\mathcal{T}(\mathcal{F})$, which satisfies all unit equational theorems of T, i.e., theorems of the form $\forall \bar{x}\ t(\bar{x}) \simeq s(\bar{x})$.

The Herbrand *interpretations* we deal with are sometimes partial, given by sets I of ground literals consistent with T. A clause C is called *T-true* in a partial interpretation I, written $I \models_T C$, if C is true in each T-model of I. Otherwise C is called *T-false* in I. This is the case when there is a T-model of I in which C is false. A ground literal L is called *T-undefined* in I if neither L nor $\overline{L}$ is T-true in I. An interpretation I is called *total* if for each ground literal, I either contains the literal or its complement.

Our restrictions on the instantiation calculus and completeness proofs are based on an ordering on closures defined as follows. First we need to adapt the

[3] If T is not a universal theory one can consider its Skolemization.

notion of a proper instantiator from [13]. We call a substitution θ a *T-proper instantiator* for a clause C if for some variable x in C, $x\theta\bot \not\simeq_T x\bot$. We will show that in our instantiation process it is sufficient to consider only T-proper instantiators. Let $\succ$ be a total simplification ordering on ground terms such that $\bot$ is a minimal term wrt. $\succ$. We assume that $\succ$ is defined on ground clauses by a total, well-founded and monotone extension of the ordering on terms as defined, e.g., in [22]. Now we lift the ordering $\succ$ from ground clauses to ground closures. Let $C \cdot \sigma$ and $D \cdot \tau$ be ground closures. We say that $C \cdot \sigma \succ' D \cdot \tau$ if either (i) $C\sigma \succ D\tau$, or (ii) $C\sigma = D\tau$ and there exists a T-proper instantiator θ for C such that $C\theta = D$. It is obvious that $\succ'$ is well-founded. We define ordering $\succ$ on closures as any total well-founded extension of $\succ'$.

3 An informal description of the instantiation procedure

Let us first informally describe our instantiation-based inference process for reasoning modulo a universal theory T.

Let S be a given set of first-order clauses. We start by mapping all variables in all clauses in S, into the distinguished constant $\bot$, obtaining a set of ground clauses $S\bot$. If $S\bot$ is T-unsatisfiable, then S is also T-unsatisfiable and we are done. Otherwise, we non-deterministically select a literal in each clause, obtaining a set of literals $\mathcal{L}$ (below we will show how this selection can be guided by information on a T-model of $S\bot$). If $\mathcal{L}$ is T-satisfiable, then S is T-satisfiable and we are done. Otherwise, we generate relevant instances of clauses from S witnessing T-unsatisfiability of $\mathcal{L}$ at the ground level. This is done based on the Unit Calculus (UC). For the refutational completeness of the overall process we need to ensure that sufficiently many instances of clauses are generated. For this we require UC to be answer-complete. Finally, we add obtained instances of clauses to S.

We prove the completeness of this instantiation process, following the steps below. First, in Section 4, we formulate an abstract calculus UC for reasoning with unit clauses, and introduce the notion of answer-completeness. Then, in Section 5, we introduce our main instantiation calculus TInst-Gen, based on an answer-complete UC, together with redundancy notions. Next, we introduce the notion of a saturated set of clauses and show that every saturated set can either be shown to be unsatisfiable by reasoning on ground clauses, or it is satisfiable (modulo the background theory). In Section 6, we show how such saturated sets can be achieved via effective fair saturation processes. We conclude with the theorem which states that every fair saturation process either stops after a finite number of steps detecting satisfiability/unsatisfiability of the initial set of clauses, or in the limit we obtain a saturated set and hence the initial set of clauses is satisfiable (modulo the background theory). In Section 7 we apply our main theorem for combining the ordered paramodulation calculus with instantiation.

4 The Unit Calculus

In this section we formulate requirements on the theory reasoner, wrt. reasoning with literals. This will be done in terms of an abstract calculus UC for proving T-unsatisfiability of sets of (selected) literals, which also provides substitutions for generating relevant instances witnessing T-unsatisfiability.

The Unit Calculus (UC)

$$\frac{L_1, \ldots, L_n}{L_1\theta, \ldots, L_n\theta}$$

where θ is such that $L_1\theta\bot \wedge \ldots \wedge L_n\theta\bot \models_T \Box$.

We assume that literals in the premise do not share variables. Let us note that the premise of UC may contain variants of the same literal.

The unit calculus will be used to generate instantiations based on T-unsatisfiable sets of literals, ensuring that the inconsistency can be detected by a theory reasoner for ground clauses.

Next we introduce the notion of answer-completeness, which is needed for overall completeness of the instantiation process. We say that UC is *answer-complete wrt.* $(T, \succ)$, if the following holds. Let $\mathcal{L} = \{L_1, \ldots, L_n\}$ be a set of literals and τ be a grounding substitution such that (i) $\mathcal{L}\tau$ is T-unsatisfiable and (ii) every proper subset of $\mathcal{L}\tau$ is T-satisfiable. Then, there is an inference in UC with $L_1, \ldots, L_n$ as a premise and $L_1\theta, \ldots, L_n\theta$ as a conclusion and a grounding substitution ρ such that $\bar{x}\tau \simeq_T \bar{x}\theta\rho$ and $\bar{x}\tau \succeq \bar{x}\theta\rho$. Let us note that an answer-complete UC calculus can produce any other instantiations, so in practice we do not need to check condition (ii) that every proper subset of $\mathcal{L}\tau$ is T-satisfiable.

Intuitively, answer-completeness requires UC to instantiate T-unsatisfiable sets of literals, with the restriction that we only need to consider instantiators generalizing minimal (wrt. $\succ$) representatives of the congruence classes defined by T. In Section 8, we show that it is possible to weaken the latter restriction further, to possibly non-minimal representatives, which is more natural in the presence of permutative subtheories.

5 The Instantiation Calculus (TInst-Gen)

In this section we introduce the instantiation calculus TInst-Gen and prove that if a set of clauses S is saturated wrt. TInst-Gen, then either $S\bot$ is already T-unsatisfiable and therefore a theory reasoner for ground clauses can detect the unsatisfiability, or S is T-satisfiable. In Section 6 we show how to achieve saturated sets.

Selection function. Our inference system will be guided by a selection function on clauses which will be based on a model for the ground clauses $S\bot$. A *selection function* sel for a set of clauses S is a mapping from clauses in S to literals such that $\mathsf{sel}(C) \in C$ for each clause $C \in S$. We say that sel is based on a model $I_\bot$ of $S\bot$, if $I_\bot \models \mathsf{sel}(C)\bot$ for all $C \in S$.

Let UC be an answer-complete wrt. $(T, \succ)$ calculus for literals and let sel be a selection function. We define TInst-Gen based on UC and sel as follows.

TInst-Gen

$$\frac{L_1 \vee C_1 \ldots L_n \vee C_n}{(L_1 \vee C_1)\theta \ldots (L_n \vee C_n)\theta}$$

where (i) the literal L_k is selected by sel in the clause $L_k \vee C_k$, for $1 \leq k \leq n$, (ii) there is an inference in UC with $L_1, \ldots, L_n$ as a premise and $L_1\theta, \ldots, L_n\theta$ as a conclusion, (iii) θ is a T-proper instantiator for at least one of $L_1, \ldots, L_n$.

Redundancy. Now we adapt the semantic notion of redundancy from [13]. Let S be a set of clauses. A ground closure C is called *T-redundant* in S if there exist closures $C_1, \ldots, C_k$ that are ground instances of S such that, (i) for each i, $C \succ C_i$, and (ii) $C_1, \ldots, C_k \models_T C$. A clause C (possibly non-ground) is called T-redundant in S if each ground closure $C \cdot \sigma$ is T-redundant in S. An inference with premises $C_1, \ldots, C_n$ and a unifier θ (thus deriving conclusions $C_1\theta, \ldots, C_n\theta$) is *$T$-redundant* in S if for any substitution ρ grounding all the $C_i\theta$ there exists an index i_0 such that $C_{i_0} \cdot \theta\rho$ is T-redundant in S.

A set of clauses S is called *TInst-saturated up to redundancy* wrt. a selection function sel if all inferences in TInst-Gen with premises from S are T-redundant in S.

Theorem 1. *Let S be a set of clauses such that $S\bot$ is T-satisfiable and* sel *be a selection based on a T-model $I_\bot$ of $S\bot$. If S is TInst-saturated up to redundancy wrt.* sel *then S is T-satisfiable.*

Proof. Suppose that S is a set of clauses such that $S\bot$ is satisfied in a T-model $I_\bot$, and sel is a selection function based on $I_\bot$. By induction over $\succ$ we construct a candidate T-model I_S for all ground instances of S. Let $C = C' \cdot \sigma$ be a ground instance of S. Suppose, as an induction hypothesis, we have defined sets of literals ϵ_D, for all ground instances D of S smaller than C wrt. $\succ$. Let I_C denote the set $\bigcup_{C \succ D} \epsilon_D$. Then, define $\epsilon_C = \{L\sigma\}$, if

1. C is T-false in I_C (i.e., there is a T-model of I_C in which C is false); and
2. L is a literal in C' such that $L\sigma$ is T-undefined in I_C (i.e., neither $I_C \models_T L\sigma$ nor $I_C \models_T \overline{L}\sigma$ holds) and $\mathsf{sel}(C') = L$.

Otherwise define $\epsilon_C = \emptyset$. In the case when $\epsilon_C = \{L\sigma\}$ we say that $L\sigma$ is *produced* by C. Finally, define I_S to be the union of all ϵ_C where C is an instance of S.

Let us first show that I_S is consistent with T. Otherwise, by compactness, there would be a finite set of literals $L_1\sigma_1, \ldots, L_n\sigma_n$ in I_S which is contradictory with T. Let $L_i\sigma$ be produced by a closure C_i for $1 \leq i \leq n$, and C_j be the maximal wrt. $\succ$ closure among them. Then, we have $I_{C_j} \models_T \overline{L}_j\sigma_j$, which contradicts the productiveness of C_j.

Let S be a set of clauses saturated under TInst-Gen and I be a total extension of I_S, consistent with T. We will show that I is a model for all ground instances of S.

First we note that our model construction is monotone: if a ground closure D is T-true in some I_C then it is T-true in all $I_{C' \succ C}$ and also true in I.

Now, by induction on $\succ$ we show that every ground instance D of S, is T-true in $I_D \cup \epsilon_D$. From this the theorem follows. Assume otherwise. Let $D = D' \cdot \sigma$ be the minimal ground instance of S that is not T-true in $I_D \cup \epsilon_D$. Let $L = \mathsf{sel}(D')$. As D is not productive and T-false in I_D we have $I_D \models_T \overline{L}\sigma$. By compactness, there is a finite set $C_1 \cdot \tau_1, \ldots, C_n \cdot \tau_n$ of closures, producing $L_1\tau_1, \ldots, L_n\tau_n$ such that $L_1\tau_1 \wedge \ldots \wedge L_n\tau_n \models_T \overline{L}\sigma$. We can assume that no proper subset of $\{L_1\tau_1, \ldots, L_n\tau_n\}$ T-implies $\overline{L}\sigma$. First we show that neither D nor any of $C_i \cdot \tau_i$ is T-redundant in S. Indeed, if $C_i \cdot \tau_i$ would T-follow from smaller closures in S, then by the induction hypothesis these closures would be T-true in $I_{C_i \cdot \tau_i}$ and hence $C_i \cdot \tau_i$ would not be productive. Similarly, if $D' \cdot \sigma$ would T-follow from smaller closures in S, it would be T-true in I_D contradicting the assumption.

It will be convenient to introduce a substitution τ which is the composition of all substitutions $\sigma, \tau_1, \ldots, \tau_n$, we assume that all clauses $D', C_1, \ldots, C_n$ are renamed apart.

Since the inference system for unit clauses UC is answer-complete wrt. $(T, \succ)$, we have that there is a UC inference with the premise $L, L_1, \ldots, L_n$ and the conclusion $L\theta, L_1\theta, \ldots, L_n\theta$ such that for a grounding substitution ρ, $\bar{x}\tau \simeq_T \bar{x}\theta\rho$ and $\bar{x}\tau \succeq \bar{x}\theta\rho$. Let us show that θ is a T-proper instantiator for at least one of $L, L_1, \ldots, L_n$. Otherwise, we would have $\bar{x}\bot \simeq_T \bar{x}\theta\bot$, where $\bar{x}$ are all variables in $L, L_1, \ldots, L_n$. This implies that $L\bot \leftrightarrow_T L\theta\bot$, $L_i\bot \leftrightarrow_T L_i\theta\bot$ for $1 \leq i \leq n$. Since $L\theta, L_1\theta, \ldots, L_n\theta$ is a conclusion of a UC inference, we have that $L\theta\bot \wedge L_1\theta\bot \wedge \ldots \wedge L_n\theta\bot \models_T \Box$ and therefore $L\bot \wedge L_1\bot \wedge \ldots \wedge L_n\bot \models_T \Box$, which contradicts that $L\bot$ and each $L_i\bot$ are true in a T-model $I_\bot$.

Since θ is a T-proper instantiator we have that TInst-Gen is applicable to $D', C_1, \ldots, C_n$, with the conclusion $D'\theta, C_1\theta, \ldots, C_n\theta$. To derive a contradiction, let us show that this inference is not redundant. For this it is sufficient to show that all closures $D' \cdot \theta\rho, C_1 \cdot \theta\rho, \ldots, C_n \cdot \theta\rho$ are not redundant. Consider $D' \cdot \theta\rho$. From $\bar{x}\tau \succeq \bar{x}\theta\rho$ and monotonicity of $\succeq$ it follows that $D' \cdot \sigma = D' \cdot \tau \succeq D' \cdot \theta\rho$. Moreover $D'\sigma \leftrightarrow_T D'\theta\rho$. Hence if $D' \cdot \theta\rho$ T-follows from smaller closures then $D' \cdot \sigma$ also T-follows from these closures, this contradicts that $D' \cdot \sigma$ is not redundant which was shown above. In the same way one can show that $C_1 \cdot \theta\rho, \ldots, C_n \cdot \theta\rho$ are not redundant.

Theorem 1 implies that if a set of clauses S is TInst-saturated up to redundancy then we can check T-satisfiability of S by checking T-satisfiability of the ground set of clauses $S\bot$. This can be done by a theory reasoner for ground clauses. In order be able to use the TInst-Gen calculus for TInst-saturation we need to show that adding the conclusion of an TInst-Gen inference to S makes the inference redundant.

Lemma 1. *Let S be a set of clauses. Consider a TInst-Gen inference, applying substitution θ and a clause $C\theta$ in the conclusion, for which θ is a T-proper instantiator (note that such a clause always exists). Then, the inference is redundant if either $C\theta$ is in S or is redundant in S. In particular, adding $C\theta$ to S makes the inference redundant.*

Proof. Let $C\theta$ be in S. Since θ is T-proper for C we have that for every ρ, $C \cdot \theta\rho \succ C\theta \cdot \rho$, and therefore $C \cdot \theta\rho$ is redundant in S. This shows that the inference is redundant. The case when $C\theta$ is redundant in S is similar.

6 Effective saturation

In this section we show how TInst-Gen saturation of a set of clauses can be achieved as a limit of a fair saturation process.

A *TInst-saturation process* is a sequence $\{\langle S^i, \mathsf{sat}_\perp^i, \mathsf{sel}^i\rangle\}_{i=1}^{\infty}$, where (i) S^i is a set of clauses, (ii) $\mathsf{sat}_\perp^i$ is a procedure for checking T-satisfiability of finite sets of ground clauses, (iii) sel^i is a selection function. In addition, we require $\{\mathsf{sat}_\perp^i\}_{i=1}^{\infty}$ and $\{\mathsf{sel}^i\}_{i=1}^{\infty}$ to satisfy some natural requirements below.

Given $\{\langle S^i, \mathsf{sat}_\perp^i, \mathsf{sel}^i\rangle\}$, a *successor state* $\{\langle S^{i+1}, \mathsf{sat}_\perp^{i+1}, \mathsf{sel}^{i+1}\rangle\}$ is obtained by one of these steps: (i) $S^{i+1} = S^i \cup N$, where N is a set of clauses such that $S^i \models_T N$; or (ii) $S^{i+1} = S^i \setminus \{C\}$, where C is TInst-redundant in S^i. If $\mathsf{sat}_\perp^{i+1}(S^{i+1}\perp)$ returns "unsatisfiable", then the process terminates with the result "unsatisfiable". Define $S^\cup = \cup_{i=1}^{\infty} S^i$. Let S^∞ denote the set of persisting clauses, that is, the lower limit of $\{S^i\}_{i=1}^{\infty}$, (i.e., $S^\infty = \cup_{i\geq 1} \cap_{j \geq i} S^j$).

In certain applications, e.g., when the theory is given as a part of the input clause set, it is natural to assume that the theory reasoner can only semi-decide T-unsatisfiability of sets of ground clauses. This is reflected in the requirements on $\{\mathsf{sat}_\perp^i\}_{i=1}^{\infty}$ and the selection functions $\{\mathsf{sel}^i\}_{i=1}^{\infty}$ below.

Requirements on $\mathsf{sat}_\perp$.

Soundness. For every finite set of clauses $S_{\mathit{fin}} \subseteq S^\cup$, we have: (i) If $\mathsf{sat}_\perp^i(S_{\mathit{fin}}\perp)$ returns "unsatisfiable" then $S_{\mathit{fin}}\perp$ is T-unsatisfiable. (ii) If $\mathsf{sat}_\perp^i(S_{\mathit{fin}}\perp)$ returns "satisfiable" then $S_{\mathit{fin}}\perp$ is T-satisfiable.

Completeness. For every finite set of clauses $S_{\mathit{fin}} \subseteq S^\infty$, we have: If $S_{\mathit{fin}}\perp$ is T-unsatisfiable then there exists i such that for all $j \geq i$, $\mathsf{sat}_\perp^j(S_{\mathit{fin}}\perp)$ returns "unsatisfiable".

Termination. For every finite subset $S_{\mathit{fin}} \subseteq S^\cup$ and every i, $\mathsf{sat}_\perp^i(S_{\mathit{fin}}\perp)$ is terminating, possibly returning "unknown".

The Requirement on sel. For a *TInst-saturation process* $\{\langle S^i, \mathsf{sat}_\perp^i, \mathsf{sel}^i\rangle\}_{i=1}^{\infty}$, it is desirable that at each step i the selection functions sel^i is based on a T-model of $S^i\perp$. Since we assume that at a step i we do not know all information about T-models of $S^i\perp$, we need a weaker requirement on the selection functions. We require that selection functions only eventually respect the models. More formally, the following should hold: either (i) some finite subset of $S^\infty\perp$ is T-unsatisfiable (hence unsatisfiability will be detected by $\mathsf{sat}_\perp^i$ for some i), or (ii) for each finite subset $S_{\mathit{fin}} \subseteq S^\infty$ there is an index i' such that for all $i \geq i'$ we have that for each $C \in S_{\mathit{fin}}$, $\mathsf{sel}^i(C)\perp$ is true in some T-model $I_\perp^i$ of $S_{\mathit{fin}}\perp$.

Next, in order to ensure that in the limit of a TInst-saturation process we always obtain a TInst-saturated set, we require the saturation process to be TInst-fair. Consider a TInst-saturation process $\{\langle S^i, \mathsf{sat}^i_\bot, \mathsf{sel}^i\rangle\}_{i=1}^{\infty}$. A TInst-Gen inference from clauses $\{L_1 \vee C_1, \ldots, L_n \vee C_n\}$ (on literals $L_1, \ldots, L_n$) in S^∞ is called *persisting* if there are infinitely many indexes i such that $\mathsf{sel}_i(L_k \vee C_k) = L_k$ for all $1 \leq k \leq n$ and conditions (i-iii) on applicability of TInst-Gen to these clauses are satisfied. A TInst-saturation process is called *TInst-fair* if every persisting TInst-Gen inference in S^∞ is redundant in S^k for some k. Let us note we can make a TInst-Gen inference redundant by simply adding the conclusion of the inference to the current clause set (see Lemma 1).

If we compare our notion of saturation to saturation in the resolution framework (e.g., [3]), then one of the key differences is that the literal selection can change at each step of the saturation. In particular, we need to consider an additional problem of showing that in the limit of a TInst-fair saturation process, we obtain a TInst-saturated set wrt. some selection function based on a T-model of the limit set of ground clauses. In the proof of the main Lemma 3 we will use the following auxiliary lemma. For a set of clauses S, let $Cl(S)$ denote the set of all ground closures in S and $\mathcal{R}(S)$ denote all redundant ground closures in S.

Lemma 2. *Let S^∞ be the set of persistent clauses of a TInst-saturation process $\{\langle S^i, \mathsf{sat}^i_\bot, \mathsf{sel}^i\rangle\}_{i=1}^{\infty}$ and $S^\cup = \cup_{i=1}^{\infty} S^i$. Then, $Cl(S^\cup)\setminus\mathcal{R}(S^\cup) = Cl(S^\infty)\setminus\mathcal{R}(S^\infty)$.*

Proof. We first show that for any set of clauses S', if a ground closure is redundant in S' then it is also redundant in $Cl(S') \setminus \mathcal{R}(S')$. Assume otherwise. Let D be a minimal (wrt. $\succ$) ground closure that is redundant in S' and not redundant in $Cl(S') \setminus \mathcal{R}(S')$. Then, D T-follows from smaller closures in S' and by the induction hypothesis each of them is either redundant in $Cl(S') \setminus \mathcal{R}(S')$ or is in $Cl(S') \setminus \mathcal{R}(S')$. This contradicts the assumption that D is not redundant in $Cl(S') \setminus \mathcal{R}(S')$.

Now we prove the statement of the lemma.

($\subseteq$) It is obvious that if a ground closure $C\cdot\sigma \in Cl(S^\cup)$ is not redundant in $S^\cup$ then it is also not redundant in S^k for any k, so $C \in S^\infty$. Since $S^\infty \subseteq S^\cup$, we have $C \cdot \sigma$ is not redundant in S^∞ and therefore $Cl(S^\cup) \setminus \mathcal{R}(S^\cup) \subseteq Cl(S^\infty) \setminus \mathcal{R}(S^\infty)$.

($\supseteq$) Assume that $Cl(S^\cup) \setminus \mathcal{R}(S^\cup) \not\supseteq Cl(S^\infty) \setminus \mathcal{R}(S^\infty)$. Let D be a ground closure in $Cl(S^\infty)\setminus\mathcal{R}(S^\infty)$ but not in $Cl(S^\cup)\setminus\mathcal{R}(S^\cup)$. We have that $D \in \mathcal{R}(S^\cup)$ and hence T-follows from smaller closures in $S^\cup$. By the statement proved above D is also T-follows from smaller closures in $Cl(S^\cup)\setminus\mathcal{R}(S^\cup)$, but by ($\subseteq$) direction these closures are also in $Cl(S^\infty) \setminus \mathcal{R}(S^\infty)$ and therefore D is redundant in $Cl(S^\infty)$. This contradicts our choice of D.

In particular, this lemma implies that if a ground closure is redundant in S^k for some k then this closure is also redundant in S^∞.

Now we are ready to show that for the limit S^∞ of a TInst-fair saturation process, such that $S^\infty\bot$ is satisfiable, we can build a T-model $I_\bot$ and a selection function sel, based on $I_\bot$ such that S^∞ is TInst-saturated wrt. sel.

Lemma 3. *Let S^∞ be a set of persistent clauses of a TInst-fair saturation process $\{\langle S^i, \mathsf{sat}^i_\perp, \mathsf{sel}^i\rangle\}_{i=1}^\infty$, and $S^\infty\perp$ is T-satisfiable. Then, there exists a T-model $I\perp$ of $S^\infty\perp$ and a selection function* sel *based on $I\perp$ such that S^∞ is TInst-saturated wrt.* sel.

Proof. Let $\{C_i\}_{i=1}^\infty$ be an enumeration of clauses in S^∞. For each n we construct a partial interpretation J^n in which all $\{C_i\perp\}_{i=1}^{i=n}$ are true and a selection function sel^n_J based on J^n (meaning that $\mathsf{sel}^n_J(C_i)\perp$ is true in J^n, i.e., true in all total consistent extensions of J^n) by induction on n. For each n the following invariants will be satisfied.

1. J^n is T-consistent and sel^n_J is a selection function for clauses $\{C_i\}_{i=1}^{i=n}$ based on J^n.
2. $J^{n-1} \subseteq J^n$ and sel^n_J coincides with sel^{n-1}_J on clauses $\{C_i\}_{i=1}^{i=n-1}$.
3. There are infinitely many k such that for a T-model $I^k_\perp$ of $S^k\perp$ we have $J^n \subseteq I^k_\perp$ and for all $1 \le l \le n$, $\mathsf{sel}^k(C_l) = \mathsf{sel}^n_J(C_l)$.

If $n = 1$, then from the Requirement on sel it follows that there exists $L \in C_1$ such that for infinitely many k, $L \in \mathsf{sel}^k(C_1)$ and $L\perp$ is true in a T-model $I^k\perp$ of $S^k\perp$. We take $J^1 = \{L\perp\}$ and $\mathsf{sel}^1_J(C_1) = \{L\}$. It is easy to see that all invariants (1–3) on J^1, sel^1_J are satisfied.

Let $n \ge 1$ and assume that we have a model J^n and sel^n_J for $\{C_i\perp\}_{i=1}^{i=n}$ such that invariants (1–3) are satisfied. Since $C_{n+1} \in S^\infty$ we have that for some m and every $p \ge m$ the following holds: (i) $C_{n+1} \in S^p$ and (ii) $\mathsf{sel}^p(C_j)\perp$ is true in a T-model $I^p_\perp$ of $\{C_i\}_{i=1}^{i=n+1}\perp$, for $1 \le j \le n+1$. From this and invariant (3) it follows that for some $L \in C_{n+1}$ there are infinitely many k such that $J^n \subseteq I^k_\perp$, and $\mathsf{sel}^k(C_l) = \mathsf{sel}^n_J(C_l)$ for all $1 \le l \le n$, and $\mathsf{sel}^k(C_{n+1}) = L$. Define $J^{n+1} = J^n \cup \{L\perp\}$ and $\mathsf{sel}^{n+1}_J(C_l) = \mathsf{sel}^n_J(C_l)$ for $1 \le l \le n$, $\mathsf{sel}^{n+1}_J(C_{n+1}) = L$. It is easy to see that all invariants (1–3) are satisfied for J^{n+1} and sel^{n+1}_J.

We define $J = \cup_{i=1}^\infty J_i$ and $\mathsf{sel}(C_i) = \mathsf{sel}^i_J(C_i)$ for $i \ge 1$. From compactness and invariants (1) and (2), it follows that J is T-consistent, and sel is a selection function based on J. We define $I_\perp$ as a total T-consistent extension of J, (note that sel is also based on $I_\perp$).

Now we need to show that S^∞ is TInst-saturated wrt. sel. Consider a TInst-Gen inference from clauses $\{L_1 \vee C_1, \ldots, L_n \vee C_n\}$ in S^∞ such that $\mathsf{sel}(L_k \vee C_k) = L_k$ for all $1 \le k \le n$. Then, from the construction of sel and in particular from the invariant (3) it follows that for infinitely many indexes i we have $\mathsf{sel}^i(L_k \vee C_k) = L_k$ for all $1 \le k \le n$. Since the TInst-saturation process is fair, this inference is redundant in S^k for some k. To show that the inference is also redundant in S^∞ it is sufficient to show that if a ground closure C is redundant in S^k for some k then C is also redundant in S^∞. This immediately follows from Lemma 2.

We summarize the obtained results in the following theorem.

Theorem 2. *Let $\{\langle S^i, \mathsf{sat}^i_\perp, \mathsf{sel}^i\rangle\}_{i=1}^\infty$, be a TInst-fair saturation process. Then, either (1) for some i the procedure $\mathsf{sat}^i_\perp(S^i\perp)$ returns "unsatisfiable" and therefore our initial set S^1 is T-unsatisfiable, or (2) for all i, $\mathsf{sat}^i_\perp(S^i\perp)$ returns either* *"unknown" or "satisfiable" and therefore, (by Lemma 3 and Theorem 1)* S^1 *is* T*-satisfiable. Moreover if for some* i, S^i *is TInst-saturated and* $\mathsf{sat}^i_\perp(S^i\perp)$ *returns "satisfiable" then at this step we can conclude that* S^1 *is* T*-satisfiable.*

Theorem 2 can be applied as follows. Assume that we have an answer-complete theory reasoner for unit clauses and a theory reasoner for ground clauses satisfying the requirements above. Then, based on the TInst-Gen calculus, we can form a TInst-fair saturation process for any set of clauses. Theorem 2 implies that this will be a complete procedure for reasoning modulo this theory. In the next section we will give an example of an application of this approach for combining the ordered paramodulation calculus with instantiation.

7 Combining instantiation with other calculi

In this section we show that the presented approach to theory reasoning is also suitable for combining the instantiation calculus with other calculi. The idea is to divide the set of input clauses into two classes: the first class can be taken as theory clauses and we apply a specialized procedure to them, the second class are the clauses treated with the instantiation calculus. In this case the theory reasoner itself can be a logical calculus which satisfies the abstract requirements introduced above.

As an example, we consider the case when theory clauses are Horn, possibly containing equality and clauses treated by instantiation contain equality only negatively, other predicates can occur positively and negatively. In order to satisfy conditions on the theory reasoner we first need an answer-complete procedure for reasoning with literals. This can be obtained based on ordered paramodulation combined with answer computations on selected literals. Such procedures, complete for answer computations, are well-studied (see [20]). Secondly, we need a procedure for theory reasoning with ground clauses. For this we can interleave ordered paramodulation with propositional reasoning, which can be done in the DPLL(T) framework [21, 25]. Let us remark that for the theory of lists and some other data structures we can use paramodulation based decision procedures for ground reasoning, studied in [1].

It is not difficult to define a TInst-fair saturation process interleaving ordered paramodulation between theory clauses, ground satisfiability checking and answer computation on selected literals with corresponding instantiation. Now we can apply Theorem 2 to show that the obtained combination of paramodulation type calculus and instantiation is complete for this class of clauses.
Remarks. Let us first note that based on our notion of redundancy we can easily justify redundancy elimination and in particular simplifications of instantiation clauses by theory clauses, such as demodulation, subsumption and T-tautology deletion. Next, we note that for answer computation we can employ other answer-complete calculi, for example calculi designed for E-unification (see [2, 10, 19]).

Now we consider the issue of a modular integration of existing reasoners for ground clauses. One of the main issues here is that usually off-the-shelf reasoners for ground clauses can reason only modulo some subtheory T' of the background theory T. (For example T' can be the theory of equality and T extends T' with some theory clauses.) Next we show how to design a reasoner for ground clauses modulo T based on a reasoner for ground clauses modulo a weaker theory T' and additional ground lemmas that can be generated by a T-reasoner for unit clauses. Such a reasoner will be sufficient for the completeness of the instantiation process modulo T.

Let S be a given set of clauses and $T' \subseteq T$. Let $\mathcal{L} = \{L_1, \ldots, L_n\}$ be a set of literals. A *T'-witness* (of T-unsatisfiability) for $\mathcal{L}\bot$ is a set of ground clauses $W = \{C_1, \ldots C_m\}$ such that (i) $S \models_T C_k$ for $1 \leq k \leq m$ and (ii) if $\mathcal{L}\bot \models_T \Box$ then $\mathcal{L}\bot \cup W \models_{T'} \Box$. In particular, if $\mathcal{L}\bot$ is T-unsatisfiable and W is a T'-witness for $\mathcal{L}\bot$, then a T' reasoner can be used to show that $\mathcal{L}\bot \cup W$ is T'-unsatisfiable. Now we formalise a saturation process based on a T' reasoner for ground clauses, assuming that we are provided with necessary T'-witnesses of T-unsatisfiability. A *TWInst-saturation process with T'-witnesses* is a sequence $\{\langle S^i, \mathsf{sat}^i_\bot, \mathsf{sel}^i, W^i\rangle\}_{i=1}^{\infty}$ such that the following holds. At each saturation step the clause set is modified as in a usual TInst-saturation. The witness set is modified as follows: $W^1 = \emptyset$ and either $W^{i+1} = W^i$ or $W^{i+1} = W^i \cup \{C_1, \ldots, C_n\}$, where $\{C_1, \ldots, C_n\}$ is a finite set of ground clauses such that $S^1 \models_T C_k$ for $1 \leq k \leq n$. If for some i, $\mathsf{sat}^i_\bot(S^i\bot \cup W^i)$ returns "unsatisfiable", then the saturation process terminates with the result "unsatisfiable". We assume that $\mathsf{sat}_\bot$, sel and W satisfy the following requirements, where we use $'$ to distinguish new requirements from the requirements on $\mathsf{sat}_\bot$ and sel in Section 6.

Requirements$'$ on $\mathsf{sat}_\bot$.

Soundness$'$. For every finite set of clauses $S_{fin} \subseteq S^\cup$, and every i we have: (i) If $\mathsf{sat}^i_\bot(S_{fin}\bot \cup W^i)$ returns "unsatisfiable" then $S_{fin}\bot \cup W^i$ is T'-unsatisfiable (and therefore T-unsatisfiable). (ii) If $\mathsf{sat}^i_\bot(S_{fin}\bot \cup W^i)$ returns "satisfiable" then $S_{fin}\bot \cup W^i$ is T'-satisfiable and W^i is a T'-witness for $\{L\bot | L = sel^i(C), C \in S_{fin}\}$ (and therefore $S_{fin}\bot$ is T-satisfiable).

Completeness$'$. For every finite set of clauses $S_{fin} \subseteq S^\infty$, we have: If $S_{fin}\bot \cup W^i$ is T'-unsatisfiable then there exists i such that for all $j \geq i$, $\mathsf{sat}^j_\bot(S_{fin}\bot \cup W^j)$ returns "unsatisfiable".

Termination$'$. For every finite subset $S_{fin} \subseteq S^\cup$ and every i, $\mathsf{sat}^i_\bot(S_{fin}\bot \cup W^i)$ is terminating, possibly returning "unknown".

The Requirement$'$ on W. Let $\{L_1 \vee C_1, \ldots, L_n \vee C_n\} \subseteq S^\infty$, and for infinitely many i we have $L_k = \mathsf{sel}^i(L_k \vee C_k)$ for $1 \leq k \leq n$. Then, for some j, W^j is a T'-witness for $\{L_1\bot, \ldots, L_n\bot\}$.

The Requirement$'$ on sel. The following should hold: either (i) for some j and some finite subset $S_{fin} \subseteq S^\infty$ we have $S_{fin}\bot \cup W^j$ is T'-unsatisfiable (hence unsatisfiability will be detected by $\mathsf{sat}^i_\bot$ for some i), or (ii) for each finite subset $S_{fin} \subseteq S^\infty$ there is an index i' such that for all $i \geq i'$ we have that for each $C \in S_{fin}$, $\mathsf{sel}^i(C)\bot$ is true in some T'-model $I^i_\bot$ of $S_{fin}\bot \cup W^i$.

Let us remark that based on a sound and complete T'-reasoner for ground clauses we can easily define $\{\mathsf{sat}^i_\bot\}_{i=1}^{\infty}$ and $\{\mathsf{sel}^i\}_{i=1}^{\infty}$ satisfying the above requirements.

In order to apply our main Theorem 2 we need to show that for a TWInst-saturation process $\{\langle S^i, \mathsf{sat}^i_\perp, \mathsf{sel}^i, W^i\rangle\}_{i=1}^{\infty}$, we have $\{\langle S^i, \mathsf{sat}^i_\perp, \mathsf{sel}^i\rangle\}_{i=1}^{\infty}$ is also a TInst-saturation process. The only nontrivial cases to check are the Completeness requirement on $\mathsf{sat}_\perp$ and the Requirement on sel. Let $S_{fin} \subseteq S^\infty$ such that $S_{fin}\perp$ is T-unsatisfiable. Let us show that in this case the TWInst-saturation process is finite. Otherwise $S_{fin} = \{L_1 \vee C_1, \ldots, L_n \vee C_n\}$, where for infinitely many i we have $L_k = \mathsf{sel}^i(L_k \vee C_k)$ for $1 \leq k \leq n$. From the Requirement$'$ on W it follows that for some j, W^j is a T'-witness for $\mathcal{L} = \{L_1\perp, \ldots, L_n\perp\}$. Therefore, $\mathcal{L}\perp \cup W^j$ is also T'-unsatisfiable which contradicts the Requirement$'$ on sel. From this it follows that the TWInst-saturation process terminates with "unsatisfiable" and therefore the Completeness requirement is satisfied. Using similar considerations we can show that the Requirement on sel is also satisfied.

We define the notion of the TWInst-fair saturation process in the same way as TInst-fair saturation. From Theorem 2 we obtain.

Corollary 1. *Let $\{\langle S^i, \mathsf{sat}^i_\perp, \mathsf{sel}^i, W^i\rangle\}_{i=1}^{\infty}$, be a TWInst-fair saturation process. Then, either (1) for some i the procedure $\mathsf{sat}^i_\perp(S^i\perp \cup W^i)$ returns "unsatisfiable" and therefore our initial set S^1 is T-unsatisfiable, or (2) for all i, $\mathsf{sat}^i_\perp(S^i\perp \cup W^i)$ returns either "unknown" or "satisfiable" and therefore, S^1 is T-satisfiable. Moreover if for some i, S^i is TInst-saturated and $\mathsf{sat}^i_\perp(S^i\perp \cup W^i)$ returns "satisfiable" then at this step we can conclude that S^1 is T-satisfiable.*

The only issue left to consider is how to generate witness sets. Let us consider the case when T' is the theory of equality and the theory reasoner for unit clauses is based on the ordered paramodulation calculus. In order to satisfy the Requirement$'$ on W, we need to ensure that if a set of literals $\mathcal{L} = \{L_1, \ldots, L_n\}$ is persistently selected and $\mathcal{L}\perp$ is T-unsatisfiable then for some i, $\mathcal{L}\perp \cup W^i$ is T'-unsatisfiable. Since the T-reasoner for unit clauses is answer-complete a proof of the empty clause will be generated from theory clauses and $\mathcal{L}$. If we propagate substitutions in such a proof from the root to the leaves (as it is done in [14] in a different context), we obtain a proof of the empty clause from the instances of the theory clauses, denoted as T_{inst}, and $\mathcal{L}$ where all substitutions map variables to variables. Therefore we have that $T_{inst}\perp \cup \mathcal{L}\perp$ is T'-unsatisfiable. Adding instantiations of the theory clauses T_{inst} to the witness set will produce the desired effect. Let us remark that clauses in the witness sets do not participate in the instantiation inferences and are only used as lemmas for the T'-reasoner on ground clauses.

Example 1. Let T' be the theory of equality. Let T extend T' with the axiom:

$$A_1 : \neg P(g(x), y) \vee f(h(x), y) \simeq g(x).$$

Let S^1 be the set of clauses:

$$C_1 : \neg \underline{P(f(x,y),c)} \vee \neg P(x,c) \qquad C_2 : \underline{P(g(x),c)} \vee \neg P(h(x),c) \qquad C_3 : \underline{P(h(x),c)}.$$

We assume that theory reasoning for unit clauses is based on the ordered paramodulation/resolution calculus wrt. the lexicographic path ordering with the precedence $P \gg f \gg h \gg g$. We assume that the ground reasoner is based on

T' and the witness set $W^1 = \emptyset$. Let us describe a possible TWInst-fair saturation process. First, we apply a ground reasoner modulo T' on $S^1\bot$ which selects $\neg P(f(x,y),c)$ in C_1, $P(g(x),c)$ in C_2, and $P(h(x),c)$ in C_3, note that $\{\neg P(f(\bot,\bot),c), P(g(\bot),c), P(h(\bot),c)\}$ is T'-satisfiable. Then we apply the reasoner for unit clauses on the set of selected literals $\{\neg P(f(x,y),c), P(g(x),c), P(h(x),c)\}$. We derive the empty clause by resolving A_1 with $P(g(x),c)$ obtaining $f(h(x),c) \simeq g(x)$, then paramodulating $f(h(x),c) \simeq g(x)$ into $\neg P(f(x,y),c)$ obtaining $\neg P(g(x),c)$ and finally resolving $\neg P(g(x),c)$ with $P(g(x),c)$. Propagating substitutions in this proof (we need to be careful to rename variables) from the root to the leaves, we obtain an instance of C_1, $C_1' : \neg P(f(h(x),c),c) \vee \neg P(h(x),c)$ and an instance of A_1, $A_1' : \neg P(g(x),c) \vee f(h(x),c) \simeq g(x)$. We add C_1' to S and $A_1'\bot$ to the witness set W. At the next step the T' reasoner for ground clauses is able to detect unsatisfiability of $S^2\bot \cup W^2$ and therefore the initial set S^1 is T-unsatisfiable.

8 Permutative Theories

In this section we show that it is possible to relax conditions on answer-completeness to appropriately accommodate reasoning modulo theories containing permutative axioms such as associativity and commutativity (AC). Let T_P be a subtheory of the theory T. Intuitively we want the unit calculus to avoid producing all permutations of witnesses equivalent wrt. T_P. For this we first need to define an appropriate T_P-compatible closure ordering. In order to ensure that such an ordering exists we impose the following restrictions on the theory T_P. Later we show that these restrictions are satisfied by permutative theories such as AC.

Condition on T_P (1). There exists a T_P-compatible simplification ordering $\succ_{T_P}$ on ground terms which is total on T_P congruence classes. An ordering $\succ_{T_P}$ is called T_P-compatible if the following holds: if $t \succ_{T_P} s$, $t \simeq_{T_P} t'$ and $s \simeq_{T_P} s'$ then $t' \succ_{T_P} s'$. We assume that $\succ_{T_P}$ is also defined on all ground clauses by an extension from ground terms, which is also a T_P-compatible T_P-total simplification ordering on ground clauses.

Now we lift the ordering $\succ_{T_P}$ from ground clauses to ground closures. For this we need an auxiliary relation $\succ'_{T_P}$ on closures. Let $C \cdot \sigma$ and $D \cdot \tau$ be ground closures. We say that $C \cdot \sigma \succ'_{T_P} D \cdot \tau$ if either (i) $C\sigma \succ_{T_P} D\tau$ or (ii) there exists a T-proper instantiator θ for C such that $C\theta = D$ and $\bar{x}\sigma \simeq_{T_P} \bar{x}\theta\tau$. Next, we would like to extend $\succ'_{T_P}$ to a well-founded order on closures. Let us show that it is not always possible for certain theories T_P. Indeed, let $\simeq_{T_P}$ be the theory of equality together with the axiom $f(f(x)) \simeq x$. Then, we would have that

$$A(f(x)) \cdot [c/x] \succ'_{T_P} A(f(f(x))) \cdot [f(c)/x] \succ'_{T_P} A(f(f(f(x)))) \cdot [c/x] \succ'_{T_P} \cdots .$$

Therefore we impose the following condition on T_P.

Condition on T_P (2): $\succ'_{T_P}$ is well-founded. Now we can define $\succ_{T_P}$ on ground closures as a T_P-total, well-founded ordering extending $\succ'_{T_P}$. Note that such an extension always exists, since any well-founded relation can be extended to a total well-founded ordering. We call $\succ_{T_P}$ a *closure ordering wrt.* T_P.

Now we show that for some important subtheories T_P, the ordering requirements on T_P can be satisfied.

Lemma 4. *Let T_P be a theory such that Condition (1) is satisfied and each T_P congruence class of ground terms is finite. Then T_P satisfies Condition (2).*

Proof. Let us show that $\succ'_{T_P}$ is well-founded. Indeed, consider an infinite chain of closures $C_1 \succ'_{T_P} C_2 \succ'_{T_P} \cdots$. Since $\succ_{T_P}$ is well-founded on ground clauses, we have that starting from some i, C_{i+k} and C_{i+k+1} satisfy the condition (ii) from the definition of $\succ'_{T_P}$, (for all $k \geq 0$). Therefore we have an infinite chain:

$$C \cdot \sigma \succ'_{T_P} C\theta_1 \cdot \sigma_1 \succ'_{T_P} C\theta_1\theta_2 \cdot \sigma_2 \succ'_{T_P} \cdots \succ'_{T_P} C\theta_1 \cdots \theta_i \cdot \sigma_i \succ'_{T_P} \cdots$$

where $\bar{x}\sigma \simeq_{T_P} \bar{x}\theta_1\sigma_1 \simeq_{T_P} \cdots \simeq_{T_P} \bar{x}\theta_1 \cdots \theta_i\sigma_i \simeq_{T_P} \cdots$. But this is impossible since each θ_i is a T-proper instantiator and therefore has a depth of at least 1 but each T_P congruence class contains only a finite number of terms.

Corollary 2. *For the theory associativity and commutativity Conditions (1-2) can be satisfied.*

Now, assume that a subtheory T_P of our background theory T satisfies Conditions (1-2). Let $\succ_{T_P}$ be a closure ordering wrt. T_P. Now we can relax restrictions on answer-completeness by considering $\succeq_{T_P}$ in place of $\succeq$. It is straightforward to check that all our previous considerations and theorems will hold in this case.

9 Conclusion

In this paper we have presented a framework for integrating theory reasoning into instantiation-based theorem proving. This integration is done in the black-box style, which allows us to integrate different theories in a uniform way. Moreover in this way we can combine different calculi with the instantiation process by treating part of the input clauses as theory clauses. We also show how in this framework it is possible to employ efficient off-the-shelf satisfiability solvers for ground clauses modulo theories. For completeness of the resulting process we impose some general requirements on the theory reasoner and show that these requirements can be naturally satisfied by calculi based on ordered paramodulation. One of our main results is the theorem which implies that if the theory reasoner satisfies the requirements then any fair instantiation process will be complete for reasoning modulo this theory. Moreover, we show how this process can be guided by (partial) information on models for ground clauses. In addition, our framework allows to justify redundancy elimination based on a semantic notion of redundant clause and redundant inference.

For future work let us mention extending our approach to theories with particular properties, like Shostak theories [12]. This would help to integrate reasoning with fragments of Arithmetic. It would also be interesting to study the relationship between our approach and hierarchical reasoning [4]. Currently the implementation of the instantiation calculus is in progress [4] which will be used to evaluate practical applicability of the proposed methods.

Acknowledgements. Many thanks to Robert Nieuwenhuis, Andrei Voronkov and anonymous reviewers for providing useful comments.

In Memoriam. It is to our deep sorrow that Professor Harald Ganzinger passed away on 3th of June 2004. His work was revolutionary in the field of automated deduction and will continue to inspire future researchers.

References

1. A. Armando, S. Ranise, and M. Rusinowitch. A Rewriting Approach to Satisfiability Procedures. *Info. and Comp.*, 183(2):140–164, June 2003.
2. F. Baader and W. Snyder. Unification theory. In A. Robinson and A. Voronkov, editors, *Handbook of Automated Reasoning*, volume I, pages 445–532. Elsevier, 2001.
3. L. Bachmair and H. Ganzinger. Resolution theorem proving. In A. Robinson and A. Voronkov, editors, *Handbook of Automated Reasoning*, volume I, pages 19–100. Elsevier, 2001.
4. L. Bachmair, H. Ganzinger, and U. Waldmann. Refutational Theorem Proving for Hierarchic First-Order Theories. *Applicable Algebra in Engineering, Communication and Computing*, 5(3/4):193–212, 1994.
5. C. Barrett, L. de Moura, and A. Stump. Design and results for the 1st satisfiability modulo theories competition. *Journal of Automated Reasoning*, 2006. to appear.
6. P. Baumgartner. FDPLL – a first-order Davis-Putnam-Logeman-Loveland Procedure. In *Proc. CADE*, volume 1831 of *LNAI*, pages 200–219, 2000.
7. P. Baumgartner and C. Tinelli. The model evolution calculus. In *Proc. CADE-19*, number 2741 in LNAI, pages 350–364. Springer, 2003.
8. P. Baumgartner and C. Tinelli. The model evolution calculus with equality. In *Proc. CADE-21*, number 3632 in LNAI, pages 392–408. Springer, 2005.
9. Peter Baumgartner. An order theory resolution calculus. In *LPAR*, volume 624 of *LNAI*, pages 119–130, 1992.
10. J. Gallier and W. Snyder. Complete sets of transformations for general E-unification. *Theoretical Computer Science*, 67(2,3):203–260, 1989.
11. H. Ganzinger, G. Hagen, R. Nieuwenhuis, A. Oliveras, and C. Tinelli. DPLL(T): Fast decision procedures. In *Poc. of CAV'04*, volume 3114 of *LNCS*, pages 175–188. Springer, 2004.
12. H. Ganzinger, T. Hillenbrand, and U. Waldmann. Superposition modulo a Shostak theory. In *CADE-19*, volume 2741 of *LNAI*, pages 182–196. Springer, 2003.
13. H. Ganzinger and K. Korovin. New directions in instantiation-based theorem proving. In *Proc. 18th IEEE Symposium on LICS*, pages 55–64. IEEE, 2003.
14. H. Ganzinger and K. Korovin. Integrating equational reasoning into instantiation-based theorem proving. In *CSL'04*, volume 3210 of *LNCS*, pages 71–84, 2004.
15. J. Hooker, G. Rago, V. Chandru, and A. Shrivastava. Partial instantiation methods for inference in first order logic. *J. of Automated Reasoning*, 28:371–396, 2002.
16. U. Hustadt, B. Motik, and U. Sattler. Reasoning in description logics with a concrete domain in the framework of resolution. In *ECAI*, pages 353–357, 2004.

[4] see http://www.cs.man.ac.uk/~korovink/iprover/

17. R. Letz and G. Stenz. Proof and model generation with disconnection tableaux. In *Proc. LPAR 2001*, volume 2250 of *LNAI*, pages 142–156, 2001.
18. R. Letz and G. Stenz. Integration of equality reasoning into the disconnection calculus. In *Tableaux 2002*, volume 2381 of *LNAI*, pages 176–190, 2002.
19. C. Lynch and B. Morawska. Goal-directed E-unification. In *Proc. of RTA'2001*, volume 2051 of *LNCS*, pages 231–245. Springer, 2001.
20. R. Nieuwenhuis. On narrowing, refutation proofs and constraints. In *Proc. of RTA'1995*, volume 914 of *LNCS*, pages 56–70. Springer, 1995.
21. R. Nieuwenhuis and A. Oliveras. Decision procedures for SAT, SAT Modulo Theories and Beyond. In *12th Int. Conf. Logic for Prog., Artif. Intell. and Reasoning (LPAR)*, volume 3835 of *LNCS*, pages 23–46, 2005.
22. R. Nieuwenhuis and A. Rubio. Paramodulation-based theorem proving. In A. Robinson and A. Voronkov, editors, *Handbook of Automated Reasoning*, volume I, pages 371–443. Elsevier, 2001.
23. D. Plaisted and Y. Zhu. Ordered semantic hyper-linking. *J. of Automated Reasoning*, 25(3):167–217, 2000.
24. M. Stickel. Automated deduction by theory resolution. *J. Autom. Reasoning*, 1(4):333–355, 1985.
25. C. Tinelli. A DPLL-based calculus for ground satisfiability modulo theories. In *Proceedings of the 8th European Conference on Logics in Artificial Intelligence*, volume 2424 of *LNAI*, pages 308–319. Springer, 2002.