Model Elimination with Basic Ordered Paramodulation

Max Moser[1]* — Christopher Lynch[2] — Joachim Steinbach[1]*

[1] Institut für Informatik, Technische Universität München
80290 München, Germany, Phone: +49-89/521096, Fax: +49-89/526502
[moser,steinbac]@informatik.tu-muenchen.de

[2] INRIA Lorraine et CRIN, Campus Scientifique, BP 101,
54602 Villers-les-Nancy cedex, France, Phone: +33-83/593012, Fax: +33-83/278319
lynch@loria.fr

Keywords. First order logic with equality, matrix methods.

1 Introduction

Essentially based on *Paramodulation* [14], various refinements for the efficient handling of equality in automated theorem proving have been proposed. The most successful among them are based on *Knuth-Bendix Completion* [8] which achieves a drastic reduction of the search space by restricting the application of equations by orderings on terms. In particular in [1, 12], refined calculi for equational clausal logic are presented, based on Paramodulation with orderings and utilizing the *Basic* strategy [7] which was originally developed for *Narrowing*.

A common property of all these approaches is that they are inherently bottom-up, which means that inferences between arbitrary clauses in general cannot be disallowed. On the other hand, a successful paradigm helping to improve search for refutations is to proceed *goal-directed*, i.e. inferences are only allowed if a so-called goal clause is involved which either leads to a new goal clause or replaces the old one. For clausal logic without equality, divers goal-directed calculi exist, one of them being the tableau-based *Model Elimination* calculus [10] which allows powerful search pruning techniques and an efficient implementation [9].

Unfortunately, it turns out that most refinements of Paramodulation are not compatible with the goal-directed paradigm. Thus, in order to preserve completeness, it is necessary to perform inferences into and even below variables (i.e. the functional reflexive axioms are needed). Since everything unifies with a variable, this is a prolific inference rule and leads to an exploding search space. This problem has been addressed in [16] by relaxing the unification to *Lazy Paramodulation*, but there is currently no such approach for Model Elimination.

Even worse, although Lazy Paramodulation restricts inferences to just non-variable positions, the pure goal-directed paradigm does not allow to take advantage of orderings. Thus, search on the clause level can be made goal-directed by giving up goal-directedness on the term level.

* Supported by the Deutsche Forschungsgemeinschaft under grant Je 112/3.

In this paper, we introduce a novel approach for goal-directed theorem proving with equality which integrates Basic Ordered Paramodulation into a Model Elimination framework. The underlying idea is to relax goal-directedness by combining the proof search with the bottom-up saturation of the original formula. In order to benefit from the goal-directedness as much as possible, only certain inferences are performed in the saturation. Actually, we show that it is enough to perform only inferences into the larger sides of positive equations. Therefore, the saturation part of the combined calculus is not complete by itself although the goal-directed part can be for certain problems. In the case without equality, the combined calculus thus reduces to pure Model Elimination, and to Completion and Narrowing in the case of unit equality.

The paper is organized as follows. Following the introduction of our combined calculus MEP in Section 3, the rest of the paper is devoted to the proof of completeness, which consists of three steps. First, in Section 4 we present a new bottom-up calculus — *Basic Factored Paramodulation* — and prove its completeness. Based on the saturation under Basic Factored Paramodulation, we derive in Section 5 abstract representations of specific unsatisfiable sets of clauses. Finally, concluding the completeness argument, in Section 6 we show that the thereby constructed abstract representations of unsatisfiable sets of clauses can be used to derive tableaux of our new calculus.

For complexity reasons, not all details of the proofs could be included in this paper. An unabridged version however can be found in [11].

2 Preliminaries

We use the standard definitions for syntax and semantics of clausal logic with equality (cf. [10, 6, 1]). For what follows, we consider the subclass of *equational formulae* where non-equational atoms A are encoded as $A \simeq \top$ for a new function symbol $\top$ and inferences among non-equality literals can be simulated by respective equality inference rules.

We also assume familiarity with rewriting (cf. [6]), orderings (cf. [17]), and constraints (cf. [12]). For what follows, reduction orderings are assumed to be ground total.

By (*ordering* and *equality*) *constraints* we mean conjunctions of *atomic* constraints over the binary predicate symbols '$\doteq$' for syntactic equality and '$\succ$' for the underlying reduction ordering. $\top$ is the empty constraint which represents *true*. Constraints are equivalently considered as sets of atomic constraints and we write $\Phi \subseteq \Theta$ if every atomic constraint in Φ is also member of Θ. A constraint Δ is *satisfiable* if there exists some ground instance $\Delta\sigma$ which is equivalent to $\top$. A *constrained clause* $C[\![\Delta]\!]$ consists of a clause C and a constraint Δ. Whenever not indicated otherwise, we rename constrained clauses to new variants before we apply inferences. When needed, we will denote constrained clauses as $\tilde{C} = C[\![\Delta]\!]$ in order to be able to distinguish them from the unconstrained disjuncts of literals C.

3 The Calculus MEP

The *Model Elimination calculus with Basic Ordered Paramodulation*, for short MEP, consists of two parts: inference rules for bottom-up saturation $\mathrm{MEP_S}$ and goal-directed tableau construction $\mathrm{MEP^T}$.

The saturation part $\mathrm{MEP_S}$ consists of only one inference rule which overlaps left-hand sides of positive equations in clauses.

Factored Positive Overlap:

$$\frac{(\, l_1 \simeq r_1 \;\vee\; \cdots \;\vee\; l_n \simeq r_n \;\vee\; B\,)[\![\Delta]\!] \qquad (\, u \simeq v \;\vee\; C\,)[\![\Phi]\!]}{(\, u[r_1]_p \simeq v \;\vee\; \cdots \;\vee\; u[r_n]_p \simeq v \;\vee\; B \;\vee\; C\,)[\![\Theta]\!]}$$

where (i) $p \in \mathcal{FP}os(u)$ and (ii) $\Theta = (\Delta \wedge \Phi \wedge u_{|p} \doteq l_1 \wedge \cdots \wedge u_{|p} \doteq l_n \wedge l_1 \succ r_1 \wedge \cdots \wedge l_n \succ r_n \wedge u \succ v)$.[3]

The result of the saturation of a formula F under $\mathrm{MEP_S}$ is denoted by $\mathcal{S}at(\mathrm{MEP_S}, F)$.

The goal-directed proof happens by constructing particular proof objects: *branches* and *tableaux*.

Definition 1. [Branches, Tableaux] A *branch* $\langle\!\langle\, L_1, L_2, \ldots, L_n \,\rangle\!\rangle$ is a sequence of literals. For a branch Ψ and a literal L_{n+1}, $\langle\!\langle\, \Psi \mid L_{n+1} \,\rangle\!\rangle$ denotes the new branch containing all literals from Ψ together with the new *leaf* literal L_{n+1}. For a branch Ψ and a clause $L_1 \vee \cdots \vee L_m$, the expression $\langle\!\langle\, \Psi \mid L_1 \vee \cdots \vee L_m \,\rangle\!\rangle$ represents the multiset of extended branches $\{\langle\!\langle\, \Psi \mid L_1 \,\rangle\!\rangle, \ldots, \langle\!\langle\, \Psi \mid L_m \,\rangle\!\rangle\}$. We write $L \in \Psi$ if Ψ contains L. A branch Ψ is said to be *closed for* a constraint Δ if Ψ contains a literal $s \not\simeq t$ such that $s\delta = t\delta$ for all substitutions δ satisfying Δ.

A *tableau* is a multiset $\mathcal{T}$ of branches together with a constraint Δ. A tableau $\mathcal{T}[\![\Delta]\!]$ is said to be *closed* if every branch in $\mathcal{T}$ is closed for Δ.[4]

The goal-directed construction of the tableau starts with the initial tableau $\langle\!\langle C \rangle\!\rangle[\![\top]\!]$ where C is any[5] clause in $\mathcal{S}at(\mathrm{MEP_S}, F)$, the saturation of the original formula under $\mathrm{MEP_S}$, and ends when the closed tableau $\emptyset[\![\Delta]\!]$ is obtained where Δ is satisfiable. To derive a closed tableau from an initial tableau, the following inference rules from $\mathrm{MEP^T}$ can be applied.

Reflection:

$$\frac{(\,\{\langle\!\langle\, \Psi \mid s \not\simeq t \,\rangle\!\rangle\} \;\cup\; \mathcal{T}\,)[\![\Delta]\!]}{\mathcal{T}[\![\Delta \wedge s \doteq t]\!]}$$

[3] $\mathcal{FP}os(u)$ is the set of non-variable positions in the term u.

[4] For convenience, we always remove closed branches since they no longer contribute to the search for a proof. Thus, a closed tableau has only an empty set of branches.

[5] In particular, we can require that C only contains negative literals. Thus, C is already contained in F.

Negative Extension/Reduction:

$$\frac{(\ \{\langle\!\langle\ \Psi \mid s \not\simeq t\ \rangle\!\rangle\}\ \cup\ \mathcal{T}\ )[\![\Delta]\!]}{(\ \langle\!\langle\ \langle\!\langle\ \Psi \mid s \not\simeq t\ \rangle\!\rangle \mid s[r]_p \not\simeq t\ \vee C\ \rangle\!\rangle\ \cup\ \mathcal{T}\ )[\![\Theta]\!]}$$

where (i) $p \in \mathcal{FP}os(s)$, (ii) $(l \simeq r \vee C)[\![\Phi]\!] \in \mathcal{S}at(\mathrm{MEP_S}, F)$ or $l \simeq r \in \Psi$, $C = \Box$ and $\Phi = \top$, (iii) $\Theta = (\Delta \wedge \Phi \wedge s_{|p} \doteq l \wedge l \succ r \wedge s \succ t)$.

Lazy Positive-to-Negative Extension/Reduction:

$$\frac{(\ \{\langle\!\langle\ \Psi \mid l \simeq r\ \rangle\!\rangle\}\ \cup\ \mathcal{T}\ )[\![\Delta]\!]}{(\ \langle\!\langle\ \langle\!\langle\ \Psi \mid l \simeq r\ \rangle\!\rangle \mid s_{|p} \not\simeq l \vee s[r]_p \not\simeq t \vee C\ \rangle\!\rangle\ \cup\ \mathcal{T}\ )[\![\Theta]\!]}$$

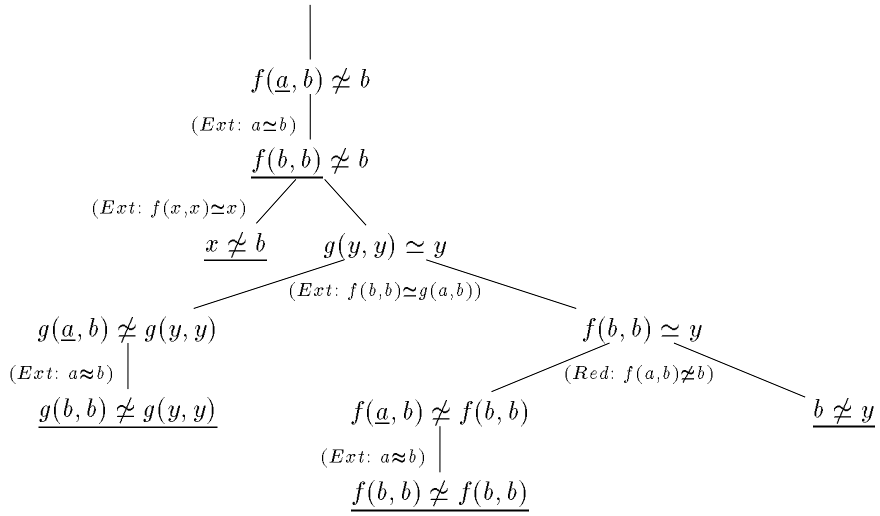
where (i) $p \in \mathcal{FP}os(s)$, (ii) $(s \not\simeq t \vee C)[\![\Phi]\!] \in \mathcal{S}at(\mathrm{MEP_S}, F)$ or $s \not\simeq t \in \Psi$, $C = \Box$ and $\Phi = \top$, (iii) $\Theta = (\Delta \wedge \Phi \wedge l \succ r)$.[6]

Lazy Positive-to-Positive Extension/Reduction:

$$\frac{(\ \{\langle\!\langle\ \Psi \mid l \simeq r\ \rangle\!\rangle\}\ \cup\ \mathcal{T}\ )[\![\Delta]\!]}{(\ \langle\!\langle\ \langle\!\langle\ \Psi \mid l \simeq r\ \rangle\!\rangle \mid v_{|q} \not\simeq l \vee u \simeq v[r]_q \vee C\ \rangle\!\rangle\ \cup\ \mathcal{T}\ )[\![\Theta]\!]}$$

where (i) $q \in \mathcal{FP}os(v)$, (ii) $(u \simeq v \vee C)[\![\Phi]\!] \in \mathcal{S}at(\mathrm{MEP_S}, F)$ or $u \simeq v \in \Psi$, $C = \Box$ and $\Phi = \top$, (iii) $\Theta = (\Delta \wedge \Phi \wedge l \succ r \wedge u \succ v)$.[6]

Example 2. The following shows a detail of a MEP-tableau for the formula $F = \{\ f(a,b) \not\simeq b,\ f(b,b) \simeq g(a,b),\ a \simeq b,\ f(x,x) \simeq x \vee g(y,y) \simeq y\ \}$. Let $\succ$ be a ground total reduction ordering such that all equations above are oriented from left to right. Notice that in this example, $\mathcal{S}at(\mathrm{MEP_S}, F) = F$.



[6] We are able to restrict the two lazy inference rules even more e.g. by replacing $s_{|p} \not\simeq l$ with $s_{|p} \not\simeq x$ and adding $x \doteq l$ for a new variable x and 'if $l \notin \mathcal{V}ar$ then $\mathcal{H}ead(s_{|p}) \doteq \mathcal{H}ead(l)$' to Θ.

The tableau starts with the only negative clause $f(a,b) \not\simeq b$. After simplifying this literal to $f(b,b) \not\simeq b$ by using $a \simeq b$ in a negative extension step, we can apply a further negative extension step into $f(x,x) \simeq x$ of $f(x,x) \simeq x \vee g(y,y) \simeq y$ which, on one hand, leads to the unifiable leaf literal $x \not\simeq b$, and on the other hand, the new positive leaf literal $g(y,y) \simeq y$. Here, the only inference we can apply is a lazy positive-to-positive extension step into the right-hand side of $f(b,b) \simeq g(a,b)$ which yields two new branches with $g(a,b) \not\simeq g(y,y)$ and $f(b,b) \simeq y$. Although $g(a,b) \not\simeq g(y,y)$ can easily be closed after applying $a \simeq b$ again, in order to close the branch with $f(b,b) \simeq y$ we need to apply a lazy positive-to-negative reduction step into the literal $f(a,b) \not\simeq b$. The rest of the tableau is closed straightforwardly.

The following theorem is the main result of this paper.

Theorem 3. [Completeness] MEP *is refutationally complete.*

In the subsequent sections, we will prove this theorem by a simulation argument, showing that for every refutation by the new bottom-up calculus Basic Factored Paramodulation BFP, there is a closed MEP-tableau using clauses in the saturation under MEP_S.

The simulation is based on the BFP calculus which is proven to be complete in Theorem 4 of Section 4, using a model construction argument similar to the one in [1]. Unfortunately, the completeness of BFP depends on the selection of maximal literals in clauses which might eventually be a negative literal although there are still other positive literals in the clause which could be overlapped.

The core of the simulation argument for the tableau construction is to 'delay' these inferences on negative literals and 'save' them by using the abstract framework of *path sets*. More precisely, for every step in the saturation under BFP we perform corresponding operations on the path sets associated with the involved clauses and thereby derive new sets of clauses with specific properties. In particular, inferences involving only positive literals are immediately performed on the clauses in the path set while inferences involving a negative literal are abstractly represented by a path and can thus be delayed until the tableau construction. Here, the Basic restriction is an important ingredient since the set of positions to which inferences can be applied is not changed by delaying them. The main result of Section 5 is Theorem 14 which guarantees the existence of specific path sets for a corresponding refutation under BFP.

Providing abstract and calculi-independent representations of contradictions of specific sets of clauses, we can use the resulting path sets in particular to derive contradictions under our goal-directed calculus. Thus, in Theorem 18 of Section 6 we show that for every such path set there is a closed MEP-tableau. For this, specific operations on path sets are defined which directly correspond with the inference rules of MEP^T.

4 Basic Factored Paramodulation

As the basis of our simulation argument, we now introduce the new *Basic Factored Paramodulation* calculus for equational clausal logic, for short BFP. Its outstanding property is that it does neither require inferences into right-hand sides of equations nor an additional rule on positive literals (*positive* or *equational factoring*), like most of the existing Basic Paramodulation calculi [1, 12, 15]. Instead, Basic Factored Paramodulation uses a related inference rule, *factored overlap*, which applies inferences on positive equations in a very homogeneous way and thus allows for a straight simulation argument.

The BFP calculus consists of *reflection* and *factored positive overlap* (cf. Section 3) together with the following inference rule.

Factored Negative Overlap:

$$\frac{(\ l_1 \simeq r_1 \ \vee \ \cdots \ \vee \ l_n \simeq r_n \ \vee \ B\ )[\![\Delta]\!] \qquad (\ s \not\simeq t \ \vee \ C\ )[\![\Phi]\!]}{(\ s[r_1]_p \not\simeq t \ \vee \ \cdots \ \vee \ s[r_n]_p \not\simeq t \ \vee \ B \ \vee \ C\ )[\![\Theta]\!]}$$

where (i) $p \in \mathcal{FP}os(s)$ and (ii) $\Theta = (\Delta \wedge \Phi \wedge s_{|p} \doteq l_1 \wedge \cdots \wedge s_{|p} \doteq l_n \wedge l_1 \succ r_1 \wedge \cdots \wedge l_n \succ r_n \wedge s \succ t)$.

In addition to the conditions for reflection and factored positive and negative overlap, we require that the literals which are involved in an inference are *selected* by an underlying *selection function*. Here, a selection function Sel is a function from the set of clauses to the power set of literals such that $Sel(C) \subseteq C$ for every clause C. Any selection function is allowed as long as for every clause C, $Sel(C)$ contains at least one negative literal or all maximal literals in C.

Theorem 4. BFP *is refutationally complete.*

Proof. We present a sketch of the completeness proof which shows how the proof in [1] based on model construction needs to be altered. There, it is shown that if a set of clauses is saturated under a set of inference rules, and it does not contain the empty clause, then a model of the set can be constructed from ground instances of the clauses. The model is a canonical set of equations generated by incrementally adding ordered instances of equations. So for each instance $C = u \simeq v \vee C'$ of a clause in the saturation we add the equation $u \simeq v$ to the model if (i) the variables in C are reduced wrt. the equations in the model from smaller instances of clauses, (ii) C is false wrt. the equations in the model from smaller instances of clauses, (iii) $u \simeq v$ is strictly maximal in C, and (iv) u cannot be reduced by the equations in the model from smaller instances of clauses. A clause that adds something to the model is called *productive*.

For our purposes, in order to take care of the missing explicit factoring inference rule, we modify step (iii) to only require that $u \simeq v$ is maximal in C.

Now we have to show that if the saturated set does not contain the empty clause, then every instance of a clause in the saturated set whose variables are reduced by the model, must be true in the model. If they are not all true, the smallest clause that is not true in the model is called a *counterexample*. It remains to show that if there is a counterexample at all, then there must be a smaller counterexample, which leads to a contradiction. For this, we distinguish among whether $Sel(C)$ contains negative literals or not. In both cases, it can be shown that $Sel(C)$ contains a literal L which is either $t \not\simeq t$ and thus allows for a reflection step or which is reducible by the model and thus, there is an factored overlap inference with a productive clause. In both cases, a smaller clause is obtained which can also be shown to be a counterexample. By the usual irreducibility arguments, the inference is lifted to the non-ground case and thus, a contradiction is derived. □

Although we don't present simplification and deletion rules, it should be noted that Basic Factored Paramodulation is compatible with rules similar to the ones presented in [1]. However, since they are not yet compatible with the path set approach, they will be excluded here.

5 Saturating Path Sets

Now we are going to introduce the abstract framework which is used to represent concrete refutations under Basic Factored Paramodulation: *path sets*.

5.1 Path Sets and Properties

Clauses and formulae have been defined as multisets. In order to be able to distinguish between the different but syntactically identical literals or clauses, we consider *sets of occurrences* of literals or clauses. Considering sets of occurrences instead of multisets, however, does not change the completeness proof of Basic Factored Paramodulation in the preceding section.

Based on sets of occurrences, we now define *path sets*, the origins of which go back to [13, 3] and have since been the basis for various proof procedures for clausal logic.

Definition 5. [Paths] Let P be a function from a set of occurrences of clauses to a set of occurrences of literals such that $P(C) \in C$ for every occurrence of a clause C. The homomorphic extension of P to sets of sets of occurrences of clauses is called a *path through* a set of occurrences of clauses.

Let P be a path through F. We write $L \in P$ if $P(C) = L$ for some $C \in F$. In particular, P is said to *select* L in C. $P|_{F'}$ is called a *subpath* of P, denoted as $P|_{F'} \subseteq P$, if P is a path through F and $P|_{F'}$ is a path through a subset F' of F such that $P|_{F'}(C) = P(C)$ for every $C \in F'$. Two paths P through F_P and Q through F_Q are said to *agree on* a set $F' \subseteq F_P \cap F_Q$ if $P|_{F'} = Q|_{F'}$.

A *path set S for F* is a set of paths through subsets of F. S' for F' is called a *path subset* of S for F if $S' \subseteq S$ and $F' \subseteq F$.

If S is a path set for a set F of occurrences of clauses, F is also often called a *matrix* [2].

Definition 6. [Operations on Path Sets] For a path set S for F and an occurrence of a clause $C \in F$, we define $S - C$ for $F \backslash \{C\}$ to be the path subset of S containing all paths from S except those selecting literals in C.

Let P and Q be paths through disjoint sets F_P and F_Q, respectively. By $P{\times}Q$ we denote the path through $F_P \cup F_Q$ which agrees with P on F_P and with Q on F_Q. For two path sets S_1 and S_2 for disjoint sets F_1 and F_2, $S_1 \times S_2$ is the path set $\{ P_1 \times P_2$ through $F_1 \cup F_2 \mid P_1 \in S_1$ through F_1 and $P_2 \in S_2$ through $F_2 \}$ for $F_1 \cup F_2$.

In what follows, we introduce particular properties of path sets which are used to describe properties of the underlying set of clauses.

Definition 7. [Covering Path Sets] A path set S is called *covering* for F if for every path P through F, there exists a path $P' \in S$ through a subset $F_{P'}$ of F such that $P' \subseteq P$. A covering path set S for F is called *minimal* if there is no proper path subset of S for F which is covering for F.

An occurrence of a clause $C \in F$ is called *essential* for a covering path set S if $S - C$ is not covering for F.

Definition 8. [Unitary Derivations/Refutations] Let M be a set of occurrences of (eventually constrained) unit literals. A sequence of negative overlap steps on M where every occurrence of a literal in M is not renamed and used only once is called a *unitary derivation of* $L[\![\Theta]\!]$ if it terminates with $L[\![\Theta]\!]$ or M contains only $L[\![\Theta]\!]$.

For a path P through a set of occurrences of clauses F, a *unitary derivation of* $s \not\simeq t[\![\Theta]\!]$ *from* P is a unitary derivation of $s \not\simeq t[\![\Theta]\!]$ from the set of occurrences of (eventually constrained) literals selected by P from the clauses in F. We say that P has a *unitary refutation with* Φ if there is a unitary derivation of $s \not\simeq t[\![\Theta]\!]$ from P such that $\Phi = (\Theta \wedge s \doteq t)$ is satisfiable. A path set S has a *unitary refutation with* Φ if Φ is satisfiable and is the union of all constraints Δ_P for which there is a $P \in S$ such that P has a unitary refutation with Δ_P. For short, we also say that S is *unitary refutable*.

5.2 Generating Path Sets by Bottom-Up Saturation

Based on the framework of path sets, we will now present inference rules on path sets which allow to derive a covering path set $S_\Box$ for a set $F_\Box$ of occurrences of clauses from $\mathcal{S}at(\mathrm{MEP_S}, F)$ such that $S_\Box$ is unitary refutable. It will be shown that these inference rules on the path sets directly correspond with the inference rules of BFP and thus, if there is a refutation under BFP, then there is a covering and unitary refutable path set. This correspondence will be described by a pair of mappings.

Definition 9. [Simulation Mappings] A *simulation mapping* from a constrained clause $\tilde{C}$ to a path set $S_{\tilde{C}}$ for $F_{\tilde{C}}$ consists of two functions:

- $\Pi_{\tilde{C}}$ assigns to each occurrence of a literal in $\tilde{C}$ a subset of $S_{\tilde{C}}$, and

- $\Gamma_{\tilde{C}}$ assigns to each occurrence of a literal in $\tilde{C}$ a subset of $F_{\tilde{C}}$.

In order to show that the final path set fulfills the desired properties, we require that for each step in the saturation, the following invariance property holds.

Definition 10. **[Conform]** Let $\tilde{C} = C[\![\Theta]\!]$ be a constrained clause. A path set $S_{\tilde{C}}$ for $F_{\tilde{C}}$ is said to *conform with* $\tilde{C}$ if
(A) $S_{\tilde{C}}$ is covering for $F_{\tilde{C}}$,
(B) for every occurrence of a literal L in C and for every path P in $\Pi_{\tilde{C}}(L)$, there is a unitary derivation of $L[\![\Phi]\!]$ such that $\Phi \subseteq \Theta$, and
(C) every path $P \in S_{\tilde{C}}$ for which there is no literal $L \in C$ with $P \in \Pi_{\tilde{C}}(L)$ has a unitary refutation with Φ such that $\Phi \subseteq \Theta$.

Initially, for every clause $\tilde{C} = C[\![\top]\!]$ in the input set F, we start with a path set $S_{\tilde{C}} = \{ P \text{ through } \{\tilde{C}\} \}$ for $F_{\tilde{C}} = \{\tilde{C}\}$. Furthermore, the corresponding simulation mapping consists of the functions
- $\Pi_{\tilde{C}}(L) = \{P\}$ for all occurrences L of literals in $\tilde{C}$ if P is a path through $\{\tilde{C}\}$ such that P selects L in $\tilde{C}$, and
- $\Gamma_{\tilde{C}}(L) = \{\tilde{C}\}$ for all occurrences L of literals in $\tilde{C}$.

The following is a straightforward consequence for these initial path sets.

Proposition 11. *For every clause in the input set, the corresponding initial path set conforms with the clause itself.*

For the following simulation, it is important to recall that for every occurrence of a positive literal L in a clause $\tilde{C}$, $|\Pi_{\tilde{C}}(L)| = 1$ and $|\Gamma_{\tilde{C}}(L)| = 1$. Furthermore, for an occurrence of a negative L, if $P, Q \in \Pi_{\tilde{C}}(L)$, then the literals selected by P and Q are either the same or variants of each other. Also, whenever we use a new occurrence $\tilde{C}$ of a constrained clause $C[\![\Theta]\!]$ for a new inference, we not only assume the variables of the constrained clause to be new variables but also that $F_{\tilde{C}}$ consists of new occurrences of constrained clauses with new variables and that $S_{\tilde{C}}$, $\Pi_{\tilde{C}}$ and $\Gamma_{\tilde{C}}$ are changed accordingly.

Based on the definitions above, we are now able to define the inference rules on path sets, each of which corresponds with an inference rule of BFP. After each definition, we will show that if the premises of the inference rules on path sets conform with respective premises of BFP, then this is also true for the relation between the conclusions.

Reflection: We consider a reflection step on $s \not\simeq t$ in $\tilde{C} = (\ s \not\simeq t \ \vee \ C'\ )[\![\Delta]\!]$ yielding $\tilde{D} = C'[\![\Theta]\!]$ with $\Theta = (\Delta \ \wedge \ s \doteq t)$.

Let $S_{\tilde{C}}$ be a path set for $F_{\tilde{C}}$ conforming with $\tilde{C}$ and $\Pi_{\tilde{C}}$, $\Gamma_{\tilde{C}}$ the accordingly adapted simulation mapping.

- For the new path set $S_{\tilde{D}}$ for $F_{\tilde{D}}$ we define
  - $S_{\tilde{D}} = S_{\tilde{C}}$, and
  - $F_{\tilde{D}} = F_{\tilde{C}}$.

- Furthermore, for the new clause $\tilde{D}$, the simulation consists of
  - $\Pi_{\tilde{D}} = \Pi_{\tilde{C}}|_{\tilde{D}}$, and
  - $\Gamma_{\tilde{D}} = \Gamma_{\tilde{C}}|_{\tilde{D}}$.

Lemma 12. *Let $S_{\tilde{C}}$ be a path set for $F_{\tilde{C}}$ which conforms with the clause $\tilde{C}$ having a satisfiable constraint. After applying a reflection step, the resulting path set $S_{\tilde{D}}$ for $F_{\tilde{D}}$ conforms with the resulting clause $\tilde{D}$ if its constraint is again satisfiable.*

Proof. Obvious by construction of $S_{\tilde{D}}$, $F_{\tilde{D}}$, $\Pi_{\tilde{D}}$ and $\Gamma_{\tilde{D}}$. □

Factored Negative Overlap: We consider a factored negative overlap step from $l_1 \simeq r_1,\dots,l_n \simeq r_n$ in $\tilde{B} = (\ l_1 \simeq r_1 \ \vee \cdots \vee \ l_n \simeq r_n \ \vee \ B'\ )[\![\Delta]\!]$ into $s \not\simeq t$ in $\tilde{C} = (\ s \not\simeq t \ \vee \ C'\ )[\![\Phi]\!]$ at $p \in \mathcal{FP}os(s)$ yielding $\tilde{D} = (\ s[r_1]_p \not\simeq t \ \vee \cdots \vee \ s[r_n]_p \not\simeq t \ \vee \ B' \ \vee \ C'\ )[\![\Theta]\!]$ where $\Theta = (\Delta \wedge \Phi \wedge s_{|p} \doteq l_1 \wedge \cdots \wedge s_{|p} \doteq l_n \wedge l_1 \succ r_1 \wedge \cdots \wedge l_n \succ r_n \wedge s \succ t)$.

Let $S_{\tilde{B}}$ for $F_{\tilde{B}}$ and $S_{\tilde{C}}$ for $F_{\tilde{C}}$ be new variants of path sets (for new occurrences of the clauses) conforming with $\tilde{B}$ and $\tilde{C}$, respectively. Furthermore, let $\Pi_{\tilde{B}}$, $\Gamma_{\tilde{B}}$ and $\Pi_{\tilde{C}}$, $\Gamma_{\tilde{C}}$ be the accordingly adapted simulation mappings. Let $S^{lr}_{\tilde{B}} = \Pi_{\tilde{B}}(l_1 \simeq r_1) \cup \cdots \cup \Pi_{\tilde{B}}(l_n \simeq r_n)$ and $F^{lr}_{\tilde{B}} = \Gamma_{\tilde{B}}(l_1 \simeq r_1) \cup \cdots \cup \Gamma_{\tilde{B}}(l_n \simeq r_n)$.

- For the new path set $S_{\tilde{D}}$ for $F_{\tilde{D}}$ we define
  - $S_{\tilde{D}} = S_{\tilde{B}} \backslash S^{lr}_{\tilde{B}} \ \cup \ S_{\tilde{C}} \backslash \Pi_{\tilde{C}}(s \not\simeq t) \ \cup \ S^{lr}_{\tilde{B}} \times \Pi_{\tilde{C}}(s \not\simeq t)$, and
  - $F_{\tilde{D}} = F_{\tilde{B}} \cup F_{\tilde{C}}$.
- Furthermore, for the new clause $\tilde{D}$, the simulation consists of
  - $$\Pi_{\tilde{D}}(L) = \begin{cases} \Pi_{\tilde{B}}(L) & \text{if} \quad L \in B' \\ \Pi_{\tilde{C}}(L) & \text{if} \quad L \in C' \\ \Pi_{\tilde{B}}(l_i \simeq r_i) \times \Pi_{\tilde{C}}(s \not\simeq t) & \text{if} \quad L = s[r_i]_p \not\simeq t \end{cases}$$

and

  - $$\Gamma_{\tilde{D}}(L) = \begin{cases} \Gamma_{\tilde{B}}(L) & \text{if} \quad L \in B' \\ \Gamma_{\tilde{C}}(L) & \text{if} \quad L \in C' \\ \Gamma_{\tilde{B}}(l_i \simeq r_i) \cup \Gamma_{\tilde{C}}(s \not\simeq t) & \text{if} \quad L = s[r_i]_p \not\simeq t \end{cases}$$

Factored Positive Overlap: We consider a factored positive overlap step from $l_1 \simeq r_1,\dots,l_n \simeq r_n$ in $\tilde{B} = (\ l_1 \simeq r_1 \ \vee \cdots \vee \ l_n \simeq r_n \ \vee \ B'\ )[\![\Delta]\!]$ into $u \simeq v$ in $\tilde{C} = (\ u \simeq v \ \vee \ C'\ )[\![\Phi]\!]$. Similar to factored negative overlap above, a simulation mapping for the resulting clause $\tilde{D}$ can be defined. The major difference is that for factored negative overlap, only modifications on paths are applied whereas for factored positive overlap, also inferences between the clauses in the corresponding sets $F_{\tilde{B}}$ and $F_{\tilde{C}}$ are performed. In particular, factored positive overlap inferences on clauses are done which corresponds to a completion and thus yields the clauses in $\mathcal{S}at(\mathrm{MEP_S}, F)$. The whole modifications on the path set can be described by two functions and are presented in detail in [11].

Lemma 13. *Let $S_{\tilde{B}}$ for $F_{\tilde{B}}$ and $S_{\tilde{C}}$ for $F_{\tilde{C}}$ be path sets which conform with the clauses $\tilde{B}$ and $\tilde{C}$, respectively and the constraints of which are satisfiable. After* *applying a factored negative or positive overlap step, the resulting path set $S_{\tilde{D}}$ for $F_{\tilde{D}}$ conforms with the resulting clause $\tilde{D}$ if its constraint is again satisfiable.*

Proof. We only present the proof for the factored negative overlap.

(A): By construction, $F_{\tilde{D}} = F_{\tilde{B}} \cup F_{\tilde{C}}$ where $F_{\tilde{B}}$ and $F_{\tilde{C}}$ are disjoint sets of occurrences of constrained clauses. Let P be any path through $F_{\tilde{D}}$. If P has a subpath in $S_{\tilde{B}} \setminus S^{lr}_{\tilde{B}}$ or $S_{\tilde{C}} \setminus \Pi_{\tilde{C}}(s \not\simeq t)$ then there is still a subpath of it also in $S_{\tilde{D}}$, by definition of $S_{\tilde{D}}$. Else, there is a subpath of it in $S^{lr}_{\tilde{B}}$ and a subpath of it in $\Pi_{\tilde{C}}(s \not\simeq t)$, i.e. there is a subpath in $S^{lr}_{\tilde{B}} \times \Pi_{\tilde{C}}(s \not\simeq t)$. Thus, by definition of $S_{\tilde{D}}$, there is again a subpath of P in $S_{\tilde{D}}$. Therefore, $S_{\tilde{D}}$ is covering for $F_{\tilde{D}}$.

(B): Obvious for $L \in B'$ and $L \in C'$ since the corresponding paths are not changed. Let $L = s[r_i]_p \not\simeq t$, $i \in [1, n]$. By assumption, for $Q \in \Pi_{\tilde{C}}(s \not\simeq t)$ there is a unitary derivation of $s \not\simeq t[\![\Psi]\!]$ such that $\Psi \subseteq \Phi$. Since $|\Gamma_{\tilde{B}}(l_i \simeq r_i)| = 1$, obviously every path in $\Pi_{\tilde{B}}(l_i \simeq r_i) \times \Pi_{\tilde{C}}(s \not\simeq t)$ allows a unitary derivation of $s[r_i]_p \not\simeq t[\![\Psi']\!]$ such that $\Psi' \subseteq \Theta$.

(C): Obvious since no new path P has been generated for which no literal in $L \in \tilde{D}$ exists with $P \in \Pi_{\tilde{D}}(L)$. $\square$

Having shown now that the inference rules on path sets preserve the property of conforming with a clause in the saturation under Basic Factored Paramodulation, we are now able, together with the proposition concerning initial path sets, to prove the main result of this section.

Theorem 14. *For every Basic Factored Paramodulation refutation of a set F there is a covering path set $S_\square$ for a set $F_\square$ of occurrences of variants of clauses from $\mathcal{S}at(\mathrm{MEP_S}, F)$ such that $S_\square$ is unitary refutable.*

Proof. By Proposition 11, for every clause $\tilde{C}$ in F, the corresponding initial path set conforms with $\tilde{C}$. Since we have a BFP refutation, all constraints of the clauses used are satisfiable and, in particular, also the constraint of the final empty clause. By Lemmata 12 and 13, for every clause $\tilde{C}$ derived in the saturation under BFP, there is a corresponding path set in the saturation under the inference rules on path sets which conforms with $\tilde{C}$. Thus, if the saturation under BFP contains the empty clause $\square[\![\Theta]\!]$, then there is a path set $S_\square$ for a set $F_\square$ which conforms with $\square[\![\Theta]\!]$. By property (A), $S_\square$ is covering for $F_\square$. Furthermore, since $\square[\![\Theta]\!]$ contains no literals and $S_\square$ for $F_\square$ conforms with $\square[\![\Theta]\!]$, we get by property (C) that every path $P \in S_\square$ has a unitary refutation. Thus, $S_\square$ is unitary refutable. Finally, by definition of the inference rules on path sets, $F_\square$ contains only occurrences of variants of clauses from $\mathcal{S}at(\mathrm{MEP_S}, F)$. $\square$

We want to point to the fact that if $S_\square$ is a covering and unitary refutable path set for $F_\square$ which conforms with $\square[\![\Theta]\!]$ and θ is a substituion satisfying Θ, the θ can be seen as a *simultaneous rigid E-unifier* of the paths in $S_\square$. For details about recent results on simultaneous rigid E-unification, cf. [5].

6 From Path Sets to MEP-Tableaux

In the preceding section, we have proven the existence of unitary refutable path sets $S_\Box$ which are covering for certain sets $F_\Box$ of occurrences of clauses. Let θ be a substitution such that $F_\Box\theta$ is ground and θ satisfies the constraints of all paths in $S_\Box$. Based on the θ-instance of $S_\Box$ for $F_\Box$, we finally show how to construct closed MEP-tableaux.

For this, with each branch of a tableau, we associate a particular path set which contains the leaf literal of the branch as an essential unit clause. Then, for each branch, applying specific operations to the associated path set leads to other path sets and therefore reflects the inference on the branch of the tableau.

There are two basic kinds of operations which we apply to path sets. On one hand, there are operations which apply inferences to literals in a path and are meant to 'solve' a path. With respect to the associated leaf literals of the branches, we have to supply this *extension* operation starting from a negative leaf literal and from a positive leaf literal. Due to the required laziness, in the case of a positive leaf literal, the operation is more complex than in the negative case. Furthermore, the lazy positive extension operation nondeterministically guarantees that at least a lazy extension step into a negative or a positive literal is possible.

On the other hand, whenever an extension step into a non-unit clause is performed, then the remaining literals in the clause need to become leaf literals of branches in the new tableau. For this, the *focusing* operation allows us to derive a corresponding path set where a whole clause in a path set is replaced by one of its literals. In particular, focusing also tells us how to obtain the path sets associated with the branches of the initial tableau.

Since the operations based on path sets in this section are performed on ground instances of clauses, positions in literals are called *basic* if they were non-variable positions before the instantiations. By applying only inferences to basic positions, we are able to lift the corresponding steps to the variable case. Furthermore, for convenience, whenever a set of units or a path has a unit refutation with some constraint, we will simply omit the reference to the constraint since we assume it to be satisfied under the current ground substitution.

Lemma 15. [Negative Path Set Extension] *Let F be a set of ground instances of occurrences of clauses containing the essential unit clause $s \not\simeq t$ and $D = l \simeq r \vee D'$ such that $s_{|p} = l$ for a basic position p in s. Let S be a covering and unitary refutable path set for F and $P \in S$ a path through some $F_P \supseteq \{s \not\simeq t,\ D\}$ such that $P(D) = l \simeq r$. Let P' be the path through $F_P \backslash \{s \not\simeq t,\ D\} \cup \{s[r]_p \not\simeq t\}$ such that P' agrees with P on $F_P \backslash \{s \not\simeq t,\ D\}$ and $P'(s[r]_p \not\simeq t) = s[r]_p \not\simeq t$.*

Then $S' = S \backslash \{P\} \cup \{P'\}$ for $F \cup \{s[r]_p \not\simeq t\}$, the path set resulting from the negative extension *from $s \not\simeq t$ into $l \simeq r$ of D on P is again covering and unitary refutable, where $s[r]_p \not\simeq t$ is essential for S'.*

For complexity reasons, we don't present the lemma for the *lazy positive extension*. The complete version can be found in [11].

Lemma 16. [Focusing] *Let F be a set of ground instances of occurrences of clauses and S a covering and unitary refutable path set for F. Let L be an occurrence of a literal in an essential clause C in F and*

S_1: *the set of all paths $Q \in S$ through F_Q where $C \notin F_Q$, and*

S_2: *the set of all paths Q' through $F_Q \backslash \{C\} \cup \{L\}$ for which there is a path $Q \in S$ through F_Q such that $Q(C) = L$, and then $Q'(L) = L$ and Q' and Q agree on $F_Q \backslash \{C\}$.*

Then $S' = S_1 \cup S_2$ for $F \backslash \{C\} \cup \{L\}$, the path set resulting from the focusing *on L in C, is again covering and unitary refutable, where L is essential for S'.*

Based on the previous lemmata, we are now able to bring the completeness proof for our MEP calculus to a close. This will be done by inductively constructing a closed MEP-tableau where the inferences on the tableau are repesented by corresponding inferences on path sets.

Lemma 17. [Ground Completeness] *For every covering and unitary refutable path set $S_\Box$ for a set $F_\Box$ of ground instances of occurrences of clauses there exists a closed* MEP-*tableau.*

Proof. W.l.o.g. assume that $S_\Box$ is minimally covering for $F_\Box$ and $F_\Box$ only contains clauses being essential for $S_\Box$. By induction, we will show how to incrementally construct a MEP-tableau. With each branch in the tableau we will associate a covering and unitary refutable path set such that the leaf literal in the branch is essential for it. By applying negative extension, lazy positive-to-negative and positive-to-positive extension and focusing on path sets, we will show how to derive new covering and unitary refutable path sets corresponding to an extended branch such that the new leaf nodes are essential in the corresponding path sets.

Let C be any occurrence of a clause in $F_\Box$ which, by assumption, is essential for $S_\Box$. By Lemma 16, for every literal in C, focusing on it yields a new covering path set where the literal focused upon is essential. Furthermore, every literal focused upon corresponds to a branch in the initial tableau.

Let $\mathcal{T}$ be any tableau containing a branch B. By induction hypothesis, there is an associated path set S^B for a set of clauses F^B which is covering and unitary refutable. Furthermore, the leaf literal L of B is essential for S^B. Since every literal in B occurs as a unit clause in F^B also L is a unit clause in F^B. We have to distinguish between whether L is (i) a negative literal or (ii) a positive literal.

(i) L is a negative literal $s \not\simeq t$: If $s = t$ then the branch is closed. Else, $s \neq t$. By Lemma 15 there is a clause C in F^B and a path P in S^B through a set containing $s \not\simeq t$ and C such that $P(C) = l \simeq r$ with $l \succ r$ and $s_{|p} = l$ at some basic position p in s such that there is a path set $S^{B'}$ which is covering and unitary refutable for $F^B \cup \{s[r]_p \not\simeq t\}$. Furthermore, the new unit clause $s[r]_p \not\simeq t$ is essential for $S^{B'}$. If $C = l \simeq r \vee C'$ is not a unit clause, then focusing applied to S^B for F^B on each literal in C' yields by Lemma 16 again covering and unitary refutable path sets where the literal focused upon is essential. Thus, for both negative extension and reduction, we obtain covering and unitary refutable path sets for the new derived branches for which the leaf literals are essential.

(ii) L is a positive literal $l \simeq r$ with $l \succ r$. According to the lemma on the positive path set extension, there is a clause D in F^B and a path P in S^B through a set containing $l \simeq r$ and D such that P selects either (a) a negative equation $s \not\simeq t$ in D with $\mathcal{H}ead(s_{|p}) = \mathcal{H}ead(l)$ at some basic position p in s, or (b) a positive equation $u \simeq v$ in D with $u \succ v$ and $\mathcal{H}ead(v_{|q}) = \mathcal{H}ead(l)$ at some basic position q in v. The remaining part of the proof is analogous with (i).

Termination: By assumption, $S_\Box$ and $F_\Box$ are finite. In order to show that we obtain a finite tableau we show that every branch generated must be finite by a measure on path sets. Let the size of a path set be the multiset of multisets of literals occurring in each path. Two sizes of path sets are compared wrt. the usual threefold reduction ordering on literals. Let S^B for F^B the path set corresponding to a branch. If we apply a negative, positive-to-positive or positive-to-negative extension step then the resulting path set(s) is (are) smaller than S^B for F^B since one path in S^B is replaced by smaller paths. Furthermore, focusing is only applied to non-unit clauses, i.e. since the paths through the neighbour literals of the focused clause are removed, the measure also decreases.

Since each branch is proven to close finitely and the tableau is only finitely branching, the tableau closes finitely. □

Theorem 18. *For every covering and unitary refutable finite path set there exists a closed* MEP*-tableau.*

Proof. Let θ be a substitution satisfying all the constraints of the paths in a covering and unitary refutable path set $S_\Box$ for $F_\Box$ such that $F_\Box\theta$ is ground. Thus, by Lemma 17 there is a closed MEP-tableau based on the θ-instance of $F_\Box$. Since in the construction of the tableau, only steps to basic positions were applied there is also a closed MEP-tableau for $F_\Box$. □

7 Conclusion

We have presented a new calculus for equational clausal logic which integrates Basic Ordered Paramodulation into a goal-directed Model Elimination framework. In order to allow the ordered application of equations in the goal-directed tableau construction, an additional bottom-up saturation phase is needed where only left-hand sides of positive equations have to be overlapped. In addition to being compatible with orderings, the calculus allows the restriction of application of equations to non-variable (in fact, basic) positions. For the completeness of the tableau part, lazy inferences are necessary for solving positive goals in the tableau, but only in a restricted form. The combined calculus can be seen as an attempt to keep the best properties of completion while only giving up part of the goal-directedness of Model Elimination. For a practical realization, of course, saturation and tableau construction are to be intertwined because saturation in general does not terminate.

The completeness of the new calculus is proven by a simulation argument which shows that for every refutation with the new saturation-based calculus Basic Factored Paramodulation there is a closed MEP-tableau. For this, path sets are used as intermediate representations of sets of clauses with specific properties. Due to being independent of the final calculus, the existence of such path sets can also be used to obtain completeness results for other goal-directed calculi.

Although for a long time, equality techniques based on orderings were requested, just recently in [4] the probably first tableau-based approach was presented. There, also an additional saturation phase is combined with the goal-directed tableau construction. The difference is, however, that in the saturation, inferences also to negative literals are allowed which might solve the underlying problem already during saturation and could be seen to be less goal-directed.

For the future, it is planned to show that the calculus is compatible with simplification and deletion rules in saturation which cannot be achieved in the current setting. Therefore, a separate model construction argument is planned which is meant to replace the simulation argument and is more flexible.

References

1. L. Bachmair, H. Ganzinger, C. Lynch, and W. Snyder. Basic Paramodulation. *Information and Computation*, 121(2):172–192, 1995.
2. W. Bibel. *Automated Theorem Proving*. Vieweg Verlag, second edition, 1987.
3. M. Davis. Eliminating the Irrelevant from Mechanical Proofs. In *Symposia of Applied Mathematics*, volume 15, pages 15–30, 1963.
4. A. Degtyarev and A. Voronkow. Equality Elimination for the Inverse Method and Extension Procedures. In *IJCAI'95*, volume 1, pages 342–347, Montreal, 1995.
5. A. Degtyarev and A. Voronkow. Simultaneous Rigid e-Unification is Undecidable. Technical Report 105, Uppsala University, 1995.
6. N. Dershowitz and J.-P. Jouannaud. Rewrite Systems. In *Handbook of Theoretical Computer Science*, pages 245–321. Elsevier Science Publishers, 1990.
7. J.-M. Hullot. Canonical Forms and Unification. In *CADE'80*, number 87 in LNCS, pages 318–334. Springer, 1980.
8. D. E. Knuth and P. B. Bendix. Simple Word Problems in Universal Algebras. In *Computational Problems in Abstract Algebra*, pages 263–297. Pergamon Press, 1970.
9. R. Letz, J. Schumann, S. Bayerl, and W. Bibel. SETHEO: A High-Performance Theorem Prover. *Journal of Automated Reasoning*, 8:183–212, 1992.
10. D. W. Loveland. *Automated Theorem Proving: A Logical Basis*. North Holland, 1978.
11. M. Moser, C. Lynch, and J. Steinbach. Model Elimination with Basic Ordered Paramodulation. Technical Report AR-95-11, Technische Universität München, 1995. http://wwwjessen.informatik.tu-muenchen.de/ftp/Automated_Reasoning/Reports/AR-95-11.ps.gz.
12. R. Nieuwenhuis and A. Rubio. Theorem Proving with Ordering and Equality Constrained Clauses. *Journal of Symbolic Computation*, 19(4):321–352, 1995.
13. D. Prawitz. An Improved Proof Procedure. *Theoria*, 26, 1960.
14. G. Robinson and L. Wos. Paramodulation and Theorem Proving in First-Order Theories with Equality. In *Machine Intelligence 4*, pages 135–150. Edinburgh University Press, 1969.
15. M. Rusinowitch. Theorem-Proving with Resolution and Superposition. *Journal of Symbolic Computation*, 11(1-2):21–50, 1991.
16. W. Snyder and C. Lynch. Goal Directed Strategies for Paramodulation. In *RTA'91*, number 448 in LNCS, pages 150–161. Springer, 1991.
17. J. Steinbach. Simplification Orderings - History of Results. *Fundamenta Informaticae, Special Issue on Term Rewriting Systems*, 1&2(24):207–247, 1995.