Deciding the E^+-class by an a posteriori, liftable order

Hans de Nivelle

February 13, 1998

Abstract

We show that the E^+-class can be decided by a liftable order, when it is applied a posteriori. This is a surprising result, because the E^+-class was one of the motivations for the study of non-liftable orders. Also surprising is the behaviour of the resolution process. Initially the variable depth of the clauses may grow, but it will not grow deeper than a certain bound. We do not make use of any type of saturation rule in our completeness proof.

1 Introduction

The E^+ class is the set of clauses, in which all literals have identical variables, and the literals are weakly covering. The class was introduced by [Tam90].
It was shown there (see also [FLTZ93]) that the following order is terminating on E^+ : $A <_v B$ iff the maximal depth at which a variable occurs in A is strictly less than the depth at which a variable occurs in B. Although resolution with this order terminates on the class E^+, it does not satisfy the condition: $A <_v B \Rightarrow A\Theta <_v B\Theta$.
Because of this the completeness of $<_v$ on E^+ was open until [Niv94a], [Niv94b], where a technique for proving completeness of non-liftable orders was introduced, called the resolution game. In fact the E^+ class was one of the motivations for its introduction.
In this paper we show that the E^+ class can be decided by a liftable order after all, when it is applied a posteriori. In [FLTZ93], pp. 109, this is posed as an open problem. (The $R_{<_d}$ refinement occurring there is equal to our $\prec$-ordering)
In [Leitsch97], there is a termination proof for the class E_1, (under the name K), which is essentially E^+ restricted to 1-variable clauses. (The E^+ class is also mentionend there, under the name K^*, on page 223. It is claimed there that saturation techniques are required)
We here prove termination for the full E^+ class, with a fully liftable order, and without saturation rules.

The behaviour of the refinement is very surprising. Initially growth of the depth (and the variable depth) is possible but only up to a certain bound.

1.1 The Resolution Calculus

We briefly introduce the resolution calculus. Contrary to the case of first order logic, we distinguish constants and variables. This makes it possible to omit the $\forall$-quantifiers in clauses.

Definition 1.1 We assume a fixed set of functions/constant symbols F, a fixed set of predicate/propositional symbols P, and a fixed set of variables V. The set of *terms* is recursively defined as follows:

1. A variable is a term.
2. If $t_1, \ldots, t_n$, with $n \geq 0$, are terms, and $f \in F$, then $f(t_1, \ldots, t_n)$ is a term.

If $t_1, \ldots, t_n$, with $n \geq 0$, are terms, and $p \in P$, then $p(t_1, \ldots, t_n)$ is an *atom*.

A *literal* is an atom A, or its negation $\neg A$. Atoms of the form A are called *positive*. Atoms of the form $\neg A$ are called *negative*.

A *clause* is a finite set of literals.

A term that contains no variables is called *ground*. A term of the form c is called *constant*. A term of the form $f(t_1, \ldots, t_n)$, with $n > 0$, is called *functional*.

Definition 1.2 We define some complexity measures for atoms/clauses/literals:

- Let A be an atom. The *depth* of A is recursively defined as follows:
  - If A is a variable, then $\text{Depth}(A) = 1$.
  - $\text{Depth}(f(t_1, \ldots, t_n))$ equals the maximum of $\{1, 1+\text{Depth}(t_1), \ldots, 1+\text{Depth}(t_n)\}$.

  The depth of a literal equals the maximal depth of its atoms. The depth of a clause c equals the maximal depth of a literal in c, or 0 for the empty clause.
- Let A be an atom. The *vardepth* of A is recursively defined as follows:
  - If A is ground, then $\text{Vardepth}(A) = -1$.
  - If A is a variable, then $\text{Vardepth}(A) = 0$.
  - Otherwise $\text{Vardepth}(f(t_1, \ldots, t_n)$ equals the maximum of $\{1 + \text{Vardepth}(t_1), \ldots, 1 + \text{Vardepth}(t_n)\}$.

The vardepth of a literal equals the vardepth of its atom. The vardepth of a clause c equals the maximal depth of a literal in c. The vardepth of the empty clause is defined as -1. The vardepth of a set of clauses C is defined as the maximal vardepth of a clause in C.

- Let A be an atom/literal/clause. $\mathrm{Var}(a)$ is defined as the set of variables that occur in A.
- Let c be a clause. $\mathrm{Varnr}(c)$ equals the size of $\mathrm{Var}(c)$. If C is a set of clauses, then $\mathrm{Varnr}(C)$ equals maximal number of variables that occurs in a clause of C.
- Let A be a literal. The *complexity* of A, written as $\#A$ equals the total number of function/constant/variable occurrences in it.

Definition 1.3 A *substitution* is a finite set of variable assignments of the form $\{V_1 := t_1, \ldots, V_n := t_n\}$, such that $V_i \neq t_i$, and $V_i = V_j \Rightarrow t_i = t_j$. The first conditions ensures non-redundancy, the second condition ensures consistency.

We write $A\Theta$ for the effect of Θ on term A.

If Θ_1 and Θ_2 are substitutions, then the *composition* of Θ_1 and Θ_2 is defined as the substitution $\{v := v\Theta_1\Theta_2| \; |v \neq v\Theta_1\Theta_2\}$.

For two literals A and B a unifier is a substitution Θ, such that $A\Theta = B\Theta$. A *most general unifier* Θ is a substitution such that $A\Theta = B\Theta$, and

$$\forall\Theta'(A\Theta' = B\Theta') \Rightarrow \exists\Sigma(\Theta' = \Theta \cdot \Sigma).$$

For the composition holds, for all terms $A(\Theta_1 \cdot \Theta_2) = (A\Theta_1)\Theta_2$.
We present an algorithm for computing the mgu. The algorithm is not efficient because its purpose is proving properties of mgu's, rather than applying it.

Definition 1.4 The following algorithm effectively decides whether or not two literals A and B have a unifier:

Let A and B be two literals, such that $A \neq B$. A *minimal difference* is obtained as follows:

1. Put $A' = A$, and $B' = B$.
2. As long as A' has form $p(t_1, \ldots, t_n)$ and B' has form $p(u_1, \ldots, u_n)$, replace A' by t_i and B' by u_i, for an i, such that $t_i \neq u_i$.

The algorithm is defined as follows: Let A and B be the terms to be unified. Put $\Theta = \{\ \}$, the idempotent substitution.

1. If $A = B$, then Θ equals the most general unifier.
2. As long as $A \neq B$, let (A', B') be a minimal difference. Then:
   - If (A', B') has form $(p(t_1, \ldots, t_n),\ q(u_1, \ldots, u_m),)$ with $p \neq q$, or $n \neq m$, then report failure.
   - If (A', B') has form (V, t), where V is a variable, $V \neq t$, but V occurs in t, then report failure.
   - If (A', B') has form (t, V), where V is a variable, $V \neq t$, but V occurs in t, then report failure.
   - If (A', B') has form (V, t) where V is a variable, and V does not occur in t, then put
     $A := A\{V := t\}$, $B := B\{V := t\}$, $\Theta := \Theta \cdot \{V := t\}$.
   - If (A', B') has form (t, V), where V is a variable, and V does not occur in t, then put
     $A := A\{V := t\}$, $B := B\{V := t\}$, $\Theta := \Theta \cdot \{V := t\}$.

Theorem 1.5 The procedure of Definition 1.4 is complete and sound, and strongly normalizing. This means that the outcome does not depend, modulo renaming substitutions, on the minimal difference selected.

Definition 1.6 We define the ordered resolution and factorization rules:
Let $\prec$ be an order on literals. The order is used to select the literals in the clauses that can be used for resolution, or factorization. Only the maximal literals in a clause can be used. There are two moments at which the ordering can be used. The first is before the substitution. This is called *a priori* application. The second is after the substitution, this is called *a posteriori* application.

APRIORI Let $\{A_1\} \cup R_1$ and $\{\neg\ A_2\} \cup R_2$, be two clauses, s.t.

- $\{A_1\} \cup R_1$ and $\{\neg\ A_2\} \cup R_2$ have no variables in common,
- A_1 and A_2 have mgu Θ,
- for no $A \in R_1$, it is the case that $A_1 \prec A$,
- for no $A \in R_2$, it is the case that $A_2 \prec A$.

Then the clause $R_1\Theta \cup R_2\Theta$ is called a *resolvent*.

Let $\{A_1, A_2\} \cup R$ be a clause, such that

- A_1 and A_2 have an mgu Θ,
- for no $A \in R$ it is the case that $A_1 \prec A$.

The clause $\{A_1\Theta\} \cup R\Theta$ is called a *factor* of $\{A_1, A_2\} \cup R$.

APOSTERIORI Let $\{A_1\} \cup R_1$ and $\{\neg\ A_2\} \cup R_2$, be two clauses, s.t.

- $\{A_1\} \cup R_1$ and $\{\neg\ A_2\} \cup R_2$ have no variables in common,

- A_1 and A_2 have mgu Θ,
- For no $A\Theta \in R_1\Theta$, it is the case that $A_1\Theta \prec A\Theta$,
- For no $A\Theta \in R_2\Theta$, it is the case that $A_2\Theta \prec A\Theta$.

Then the clause $R_1\Theta \cup R_2\Theta$ is called a *resolvent*.
Let $\{A_1, A_2\} \cup R$ be a clause, such that

- A_1 and A_2 have mgu Θ,
- for no $A\Theta \in R\Theta$ is it the case that $A_1\Theta \prec A\Theta$.

Then the clause $\{A_1\Theta\} \cup R\Theta$ is called a *factor* of $\{A_1, A_2\} \cup R$.

It is not completely trivial to find an example where a priori and a posteriori application of the order differ, in the case that the order is liftable.

Example 1.7 Let $\prec$ be (a sort of) a Knuth-Bendix order: $A \prec B$ if $\# A < \# B$ and each variable occurring in A occurs at least as often in B.
The following derivations are allowed by a priori application of $\prec$, but they are blocked by a posteriori application:
Resolve $\{p(X, X, Y), q(X, Y, Y)\}$ with $\{\neg\, p(0, 0, s(s(0)))\}$.
Resolve $\{p(X, X, Y, 0), q(X, Y, Y, s(s(0))))\}$ with $\{\neg\, p(0, 0, Y, 0)\}$.

Definition 1.8 Let $C = \{c_1, \ldots, c_n\}$ be a clause set. We call C *unsatisfiable* if its first order meaning is unsatisfiable. We call C *propositionally unsatisfiable* if for *every* substitution Θ, the set $C\Theta$ is unsatisfiable.

Theorem 1.9 Let $\prec$ be an order satisfying

$$A \prec B \Rightarrow A\Theta \preceq B\Theta.$$

(the *liftability* condition). Then $\prec$-ordered resolution with factoring is complete.

In fact the order need not be fully liftable. It is sufficient if A and B are restricted to literals that can occur in a derivation, and $A\Theta$ and $B\Theta$ are restricted to literals that can occur in an unsatisfiable instance.
This makes the following more complicated condition:

Definition 1.10 Let $\prec$ be a relation on literals. We call $\prec$ liftable if it satisfies the following conditions:

For every unsatisfiable clause set C, there is a propositionally unsatisfiable set $\overline{C}$ of instances of C, and an order $\prec_g$ on the literals in $\overline{C}$, s.t.

1. for all A and B with $A \prec B$, that can occur together in a clause that is $\prec$-derivable from C, and

2. for all $A\Theta$ and $B\Theta$ that occur in $\overline{C}$, (not necessarily in the same clause)

it is the case that

$$A\Theta \prec_g B\Theta, \text{ or } A\Theta = B\Theta.$$

So $\prec$ need not even be an order by itself, but $\prec_g$ has to be an order.

2 Covering Literals

In order to obtain termination for resolution it is necessary that the literals in the clause sets have a special form: The must be *weakly covering*. Although the definition of this notion is technical, covering literals arise naturally from the Skolemization of decidable classes. We show that covering literals are preserved under resolution, and that the mgu of two covering literals is not larger than the largest of them.

Definition 2.1

- A literal A is *covering* if every functional subterm t of A contains all variables of A.
- A literal A is *weakly covering* if every functional, non-ground subterm t of A contains all variables of A.

Covering and weakly covering literals are typically the result of skolemization, when the prefix ends in an existential quantifier. If an atom $a(\overline{x}, y)$ with in the scope of $\forall \overline{x} y$ is skolemized the result equals $a(\overline{x}, f(\overline{x}))$, which is covering. If $a(\overline{x}, y)$ contains ground terms, then the result is weakly covering.
The main property of (weakly) covering literals is that they do not grow when they are unified. Before we prove this we need a technical property.

Definition 2.2 Let Θ be a substitutution. Θ is a *weak renaming* if for each variable V, the result $V\Theta$ is either ground, or a variable.

So what is excluded are substitutions of the type $\{X := f(X)\}$.

Lemma 2.3 Let A and B be two weakly covering literals, such that $\text{Var}(A) \subseteq \text{Var}(B)$, and $\text{Vardepth}(A) \leq \text{Vardepth}(B)$. Let Θ be the mgu of A and B. Then $\text{Vardepth}(A) = \text{Vardepth}(B)$.

proof:
Suppose that $\text{Vardepth}(A) \neq \text{Vardepth}(B)$. Then $\text{Vardepth}(A) < \text{Vardepth}(B)$. For every substitution Θ, it follows $\text{Vardepth}(A\Theta) < \text{Vardepth}(B\Theta)$, because every variable of A occurs in the deepest term of B. But then A and B cannot

be unified.
end of proof

The following states that when two weakly covering terms are unified, the result is not larger than the largest of them.

Theorem 2.4 Let A and B be weakly covering literals that have an mgu Θ. Let $C = A\Theta = B\Theta$. Then:

1. C is weakly covering,
2. $\text{Vardepth}(C) \leq \text{Vardepth}(A)$, or $\text{Vardepth}(C) \leq \text{Vardepth}(B)$.
3. $\text{Varnr}(C) \leq \text{Varnr}(A)$ or $\text{Varnr}(C) \leq \text{Varnr}(B)$.

proof:
Without loss of generality we consider two cases:

- $\text{Vardepth}(A) = \text{Vardepth}(B)$. Assume that the mgu Θ is constructed by the algorithm of Definition 1.4. We will show that for every variable X, the result $X\Theta$ is ground, or a variable.

  Suppose there were a minimal difference of the form (X, t) (or converse), where t is non-ground, and functional. t has to contain all variables of B, since B is weakly covering. The variable X cannot occur in B, because of the occurs check. Define $\Theta' = \{X := t\}$. We have $\text{Var}(B\Theta') \subseteq \text{Var}(A\Theta')$, and $\text{Vardepth}(B\Theta') < \text{Vardepth}(A\Theta')$. This contradicts Lemma 2.3. By following the iteration of Definition 1.4 it follows that the unifier Θ does not assign a non-ground functional term to any of the variables in A or B.

- $\text{Vardepth}(A) < \text{Vardepth}(B)$. Assume that the mgu Θ is constructed by the algorithm of Definition 1.4.

  Let (t_1, t_2) be a minimal difference based on a term with maximal vardepth in B. (t_1, t_2) must be of the form (X, t), where X does not occur in t, and t is non-ground, functional, and contains all variables of B. Put $\Theta' = \{X := t\}$.

  We have $\text{Var}(B\Theta') \subseteq \text{Var}(A\Theta')$ and $\text{Vardepth}(B\Theta') < \text{Vardepth}(A\Theta')$. By Lemma 2.3 it follows that $\text{Vardepth}(A\Theta') = \text{Vardepth}(B\Theta')$.

end of proof

The following is necessary in order to show that when two clauses resolve, and they consist of weakly covering literals, that then the resolvent consists of weakly covering literals. Let $R_1\Theta \cup R_2\Theta$ be a resolvent of $\{A_1\} \cup R_1\}$ and $\{\neg\, A_2\} \cup R_2$. Theorem 2.4 states that $A_1\Theta$ is weakly covering, but we have proven nothing about the literals in $R_i\Theta$.

Theorem 2.5 Let A and B be literals and let Θ be a substitution such that

1. $\mathrm{Var}(A) \subseteq \mathrm{Var}(B)$,
2. A is weakly covering,
3. $B\Theta$ is weakly covering.

Then $A\Theta$ is weakly covering.

proof:
Let t_2 be a non-ground, functional subterm of $A\Theta$. Let V_2 be a variable of $A\Theta$. We have to show that V_2 occurs in t_2.
There must exist a variable V_1, that occurs in A, and such that $V_1\Theta$ contains V_2. (If V_2 was not introduced by Θ, then $V_1 = V_2$.) There are two possibilities:

1. There is a $t_1[W]$ in A, such that $t_1[W\Theta] = t_2$.
   In that case $t_1[W]$ is non-ground, and functional. because of this $t_2 = t_1[W\Theta]$ contains V_1.
2. There is a W in A, such that t_2 is a subterm of $W\Theta$. (This case includes $t_2 = W\Theta$)
   Because W also occurs in B, the term $B\Theta = t_2$ occurs in $B\Theta$. Because $B\Theta$ is weakly covering, V_2 must occur in t_2.

end of proof
Usually ([FLTZ93]) this theorem is found with a 4th condition: B is weakly covering. We have shown that this condition can be dropped. That this results in a true generalization can be seen by putting: $A = p(X,Y), B = p(X, s(Y)), \Theta = \{Y := s(X,Y)\}$. B is not weakly covering, but $A\Theta$ is.
No more conditions can be dropped from Theorem 2.5, as can be seen from the following examples:

1. $A = p(X,Y),\ B = p(X),\ \Theta = \{X := s(X)\}$.
2. $A = p(X, s(Y)),\ B = p(X,Y),\ \Theta = \{\}$.
3. $A = p(X,Y),\ B = p(X,Y),\ \Theta = \{X := s(X)\}$.

The following states that the side literals can be bound by the literals resolved upon.

Lemma 2.6 If

1. $\mathrm{Var}(A) \subseteq \mathrm{Var}(B)$,
2. A is weakly covering,

3. B is weakly covering,

4. $\text{Vardepth}(A) \leq \text{Vardepth}(B)$,

Then: $\text{Vardepth}(A\Theta) \leq \text{Vardepth}(B\Theta)$.

There is another technical fact that we are going to need, namely that unification does not introduce new ground terms in weakly covering literals, unless the result is completely ground.

Lemma 2.7 Let $C = A\Theta = B\Theta$ be the most general unifier of two weakly covering literals. If C is not ground by itself, then every ground term of C occurs either in A or in B.

proof:
If either A or B is ground, then C will be ground, so let us consider the case where neither A, nor B is ground.
Assume that Θ is obtained by the algorithm of Definition 1.4. Then we can write

$$\Theta = \Sigma_1 \cdot \ldots \cdot \Sigma_n,$$

where each Σ_i is a simple substitution of the form $\{X := t\}$.
We show by induction that if both $A\Sigma_1 \cdots \Sigma_i$ and $B\Sigma_1 \cdots \Sigma_i$ are non-ground, then they do not contain new ground terms. This is clearly true for $i = 0$. Suppose it is true for some i. Then $\Sigma_{i+1} = \{X := t\}$, where t is a term occurring either in $A\Sigma_1 \cdots \Sigma_i$, or $B\Sigma_1 \cdots \Sigma_i$. Hence there is no ground term, not in A or B, in t.
This means that the only manner in which a new ground term can be obtained is when t is ground, and X was the last variable in a functional term. Now if variable X occurs somewhere in a functional term, then this term either contains all variables of $A\Sigma_1 \cdots \Sigma_i$, or of $B\Sigma_1 \cdots \Sigma_i$. (Consequence of the proof of Theorem 2.4) Hence if the last variable is substituted away, either $A\Sigma_1 \cdots \Sigma_i$, or $B\Sigma_1 \cdots \Sigma_i$ is ground. Hence one of $A\Theta$ or $B\Theta$ is ground, and so they are both ground.
end of proof

3 The E^+-Class

We are now ready to introduce the E^+-class.

Definition 3.1 Let C be a clause set. C is in the class E^+ if for every clause $c \in C$ the following holds:

1. For all $A, B \in c$, it is the case that $\text{Var}(A) = \text{Var}(B)$.

2. All literals $A \in c$ are weakly covering.

Sometimes the E^+-class is defined in an extended manner in which the clauses are allowed to consist of more than one component. This is not relevant for our discussion.
It was already known that the E^+-class is decidable, and that it can be decided by the following order, when it is applied a priori:

Definition 3.2 Let A and B be two literals. $A <_v B$ if $\text{Vardepth}(A) < \text{Vardepth}(B)$.

This order is non-liftable, which makes it difficult to prove completeness, but proving termination is fairly easy. With the liftable order $\prec$ that we will introduce, the picture is exactly reversed. Proving termination is difficult, but the completeness is a standard result.
That the order $<_v$ is non-liftable, can be seen from the following: We have

$$p(X, s(0)) <_v p(s(X), 0), \text{ and } p(s(0), X) <_v p(0, s(X)).$$

Applying $\{X := 0\}$ we obtain:

$$p(0, s(0)) <_v p(s(0), 0), \text{ and } p(s(0), 0) <_v p(0, s(0)),$$

which contradicts the fact that $<_v$ is an order.
Showing that $<_v$ cannot satisfy Definition 1.10 is slightly more involved. We need to show that the previous conflict can be enforced by a clause set: Take the clause set

$$C = \{\{p(0, s(X))\}, \{\neg\, p(X, s(0)), p(s(X), 0)\}, \{\neg\, p(s(0), X)\}\}$$

C is unsatisfiable, and has only one unsatisfiable set of ground instances:

$$C_g = \{\{p(0, s(0))\}, \{\neg\, p(0, s(0)), p(s(0), 0)\}, \{\neg\, p(s(0), 0)\}\}$$

Since all the ground instances of the previous example lie in C_g, and there is no other unsatisfiable set of ground instances, the order $<_v$ violates Definition 1.10.

Theorem 3.3 Let C be a set of clauses in E^+. Every clause that can be derived from C, (by unrestricted resolution, or factoring) is also in E^+.

proof:
Follows from Theorem 2.4 and Theorem 2.5.
end of proof

Definition 3.4 We define the following order on literals: $A \prec B$ if

1. $\text{Vardepth}(A) < \text{Vardepth}(B)$ and $\text{Depth}(A) < \text{Depth}(B)$.
2. $\text{Vardepth}(A) = \text{Vardepth}(B) = -1$, and $\text{Depth}(A) < \text{Depth}(B)$.

Order $\prec$ is not liftable in the strong sense of Theorem 1.9, but is satisfies the conditions of Definition 1.10. This order is called $<_d$ in [FLTZ93], page 106.

Theorem 3.5 Relation $\prec$ is liftable.

proof:
We show that $\prec$ satisfies Definition 1.10. Let C be an unsatisfiable clause set in E^+. The class E^+ is closed under unrestricted resolution with factoring. Because of this we may assume that A and B are weakly covering, and $\text{Var}(A) = \text{Var}(B)$. Take for $\prec_g$ the following order:

$$A\Theta \prec_g B\Theta \text{ iff } \text{Depth}(A\Theta) < \text{Depth}(B\Theta).$$

Suppose that $A \prec B$. There are two cases:

1. Both A and B are ground. Then $A \prec B$ because of the second case of the definition. The substitution does not affect A and B, and hence $A\Theta \prec_g B\Theta$.
2. Both A and B are non-ground. Then $A \prec B$ because of the first case of the definition. We need to show that $\text{Depth}(A\Theta) < \text{Depth}(B\Theta)$. Let $d = \text{Depth}(A\Theta)$. There are two cases to consider:
   (a) The depth d occurs in a term (which is a constant or variable) created by Θ. Then since $\text{Var}(A) = \text{Var}(B)$, $\text{Vardepth}(A) < \text{Vardepth}(B)$, the same term is created by Θ in B, and at a deeper position. Hence $\text{Depth}(A\Theta) < \text{Depth}(B\Theta)$.
   (b) The depth d occurs in a term not created by Θ. Then the term occurred already in A. Because of $\text{Depth}(A) < \text{Depth}(B)$ there is a deeper term in B, and also in $B\Theta$.

end of proof

The following also holds, although it is irrelevant for completeness:

Lemma 3.6 Relation $\prec$ is an order within a clause.

The termination proof for the $<_v$ order is based on the fact that it is impossible to derive a clause c, with $\text{Vardepth}(c) > \text{Vardepth}(C)$, or containing more variables than a clause of C. This, together with Lemma 2.7 ensures termination. So it is possible that $\text{Depth}(c) > \text{Depth}(C)$, for a derived clause, but this is harmless, as this can be caused only by a finite set of ground terms.

Example 3.7 Put $c_1 = \{p(X, Y, s(f(X, Y))), q(X, Y, s(f(X, Y)))\}$, and $c_2 = \{\neg\ p(s(s(0)), X, Y\}$. The resolvent $c = \{p(s(s(0)), Y, s(f(s(s(0)), Y)))\}$ is allowed. We have $\text{Vardepth}(c) \leq \text{Vardepth}(c_1)$, $\text{Varnr}(c) \leq \text{Varnr}(c_1)$, but $\text{Depth}(c) > \text{Depth}(c_1)$, $\text{Depth}(c_2)$. The ground term $s(s(0))$ occurred already in c_2.

This is not true for the a posteriori $\prec$-order, as can be seen from the following example:

Example 3.8 Put $c_1 = \{\neg\ p(X, s(s(s(s(0))))), p(X, s(s(s(s(0)))))\}$. Resolving the clause twice with itself is allowed, and results in
$c = \{\neg\ p(X, s(s(s(s(0))))), p(s(s(s(s(X)))), s(s(s(s(0)))))\}$, with $\text{Vardepth}(c) = 4$, and $\text{Vardepth}(c_1) = 1$.

We will show however that it is not possible to continue this initial growth. So this unpleasant behaviour can only occur in clauses with low depth.

Definition 3.9 Let A and B be literals. We define the following notions:

1. $A \sqsubseteq B$ iff

$$\text{Depth}(A) - \text{Vardepth}(A) \leq \text{Depth}(B) - \text{Vardepth}(B).$$

2. $A \equiv B$ iff $A \sqsubseteq B$ and $B \sqsubseteq A$.

We have $p(X, s(0)) \sqsubseteq p(X, s(s(0)))$, but not $p(X, s(0)) \sqsubseteq p(X, 0)$.
We have the following technical fact:

Lemma 3.10 Let A and B weakly covering literals with $\text{Var}(A) = \text{Var}(B)$. Let Θ be a substitution, such that $A\Theta$ is non-ground. Then

$$\text{Vardepth}(A\Theta) - \text{Vardepth}(B\Theta) = \text{Vardepth}(A) - \text{Vardepth}(B).$$

proof:
Because the deepest term in both A and B must contain all variables.
end of proof

Theorem 3.11 The relation $\sqsubseteq$ satisfies the following conditions:

O1 Always $A \sqsubseteq A$.

O2 $A \sqsubseteq B$ and $B \sqsubseteq C$ implies $A \sqsubseteq C$.

L1 For weakly covering A and B, with $\text{Var}(A) = \text{Var}(B)$, for substitutions Θ, such that $A\Theta$ is non-ground,

$$A \sqsubseteq B \Rightarrow A\Theta \sqsubseteq B\Theta.$$

L2 For weakly covering A and B, with $\text{Var}(A) = \text{Var}(B)$, for substitutions Θ, such that $A\Theta$ is non-ground, if $\text{Depth}(A\Theta) > \text{Depth}(A)$, then

$$A\Theta \sqsubseteq B\Theta.$$

proof:

O1 Because

$$\text{Depth}(A) - \text{Vardepth}(A) = \text{Depth}(A) - \text{Vardepth}(A).$$

O2 Evidently

$$\text{Depth}(A) - \text{Vardepth}(A) \leq \text{Depth}(B) - \text{Vardepth}(B),$$

and

$$\text{Depth}(B) - \text{Vardepth}(B) \leq \text{Depth}(C) - \text{Vardepth}(C),$$

implies

$$\text{Depth}(A) - \text{Vardepth}(A) \leq \text{Depth}(C) - \text{Vardepth}(C),$$

L1 Assume that $A \sqsubseteq B$, so

$$\text{Depth}(A) - \text{Vardepth}(A) \leq \text{Depth}(B) - \text{Vardepth}(B).$$

We want to show that

$$\text{Depth}(A\Theta) - \text{Vardepth}(A\Theta) \leq \text{Depth}(B\Theta) - \text{Vardepth}(B\Theta).$$

Write $k = \text{Vardepth}(B) - \text{Vardepth}(A) = \text{Vardepth}(B\Theta) - \text{Vardepth}(A\Theta)$.

Note that k is possibly negative. Let $d = \text{Depth}(A\Theta)$. We must show that $\text{Depth}(B\Theta) \geq d + k$. Let T be a deepest variable or constant in $A\Theta$. There are two cases:

- T was introduced by Θ. In that case T was also introduced in $B\Theta$, and T causes there a depth of at least $d + k$.
- T was already present in A. In that case $\text{Depth}(A\Theta) = \text{Depth}(A) = d$, and all terms introduced by the substitution Θ had a depth not larger than $\text{Depth}(A) - \text{Vardepth}(A)$. Because of $A \sqsubseteq B$ it must be the case that $\text{Depth}(B) \geq d + k$.

L2 Write $k = \text{Vardepth}(B) - \text{Vardepth}(A) = \text{Vardepth}(B\Theta) - \text{Vardepth}(A\Theta)$. Let T be a deepest variable or constant in $A\Theta$. Let $d = \text{Depth}(T)$. We have to show that $\text{Depth}(B\Theta) \geq d + k$. Because $\text{Depth}(A\Theta) > \text{Depth}(A)$ this T must have been introduced by the substitution. Because of this T also occurs in $B\Theta$. Then $B\Theta$ must have a depth of at least $d + k$.

end of proof

Property **L1** does not hold when the substitution Θ makes the literals ground. The following example demonstrates this: We have $p(s(X), s(0)) \sqsubseteq p(X, s(0))$, but substitution $\Theta = \{X := s(s(s(0)))\}$ results in $p(s(s(s(s(0)))), s(0)) \not\sqsubseteq p(s(s(s(0))), s(0))$.

Lemma 3.12 Let C be a clause set in E^+. Let k be an integer with $k > \text{Depth}(C)$. For every non-ground clause c that is derivable from C with $\prec$ applied a posteriori, the following holds:

1. If $A, B \in c$, and $\text{Depth}(A),\ \text{Depth}(B) > k$, then $A \equiv B$.
2. If $A, B \in c$, with $\text{Depth}(A) > k$, and $\text{Depth}(B) \leq k$, then $A \sqsubseteq B$.

proof:
We may assume that all derived clauses are in E^+. Initially (1) and (2) are trivially satisfied.
It is sufficient to show that properties (1) and (2) are preserved by substitution, deletion of a literal, and $\prec$-ordered propositional resolution, as factorization and resolution, using $\prec$ a posteriori can be decomposed into these rules.

- Deletion of a literal.
  Let c' be obtained from c by deleting one literal. Whenever $A, B \in c'$ also $A, B \in c$. Because of this A and B have the desired properties.

- Propositional resolution.
  Let c be a $\prec$-ordered, propositional resolvent of the clauses $c_1 = \{A\} \cup R_1$ and $c_2 = \{\neg\, A\} \cup R_2$.

  If both c_1 and c_2 contain no literals with depth $> k$, then also c contains no literals with depth $> k$. Because of this c trivially satisfies (1) and (2).

  If one of c_1, c_2 contains a literal with depth $> k$, then $\text{Depth}(A) > k$. This can be seen from the following:

  Without loss of generality assume that c_1 contains a literal B with $\text{Depth}(B) > k$, but that $\text{Depth}(A) \leq k$. Then evidently $\text{Depth}(A) < \text{Depth}(B)$. Since $B \sqsubseteq A$, by (2), it must be the case that $\text{Vardepth}(A) < \text{Vardepth}(B)$. But then we have $A \prec B$, which is a contradiction, since B would block A.

  Let B_1, B_2 be literals with $\text{Depth}(B_1),\ \text{Depth}(B_2) > k$. If B_1 and B_2 originate together from R_1, or R_2, then (1) and (2) are trivially inherited. So assume, without loss of generality that B_1 originates from R_1 and B_2 originates from R_2. Since $B_1 \equiv A$, and $B_2 \equiv A$, by induction, it must be the case that $B_1 \equiv B_2$.

  Let B, C be literals with $\text{Depth}(B) > k$, and $\text{Depth}(C) \leq k$. Without loss of generality assume that B originates from R_1, and that B_2 orginates from R_2. Then by induction we have $A \equiv B$ and $A \sqsubseteq C$. It follows that $B \sqsubseteq C$.

- Substitution.
  Let Θ be a substitution not making c ground. Let $A\Theta,\ B\Theta$ be a pair of literals in $c\Theta$. Without loss of generality consider the following cases. The other cases are either impossible, or symmetric to one of the cases we consider:

  1. Both $\text{Depth}(A),\ \text{Depth}(B) \leq k$, and both $\text{Depth}(A\Theta),\ \text{Depth}(B\Theta) \leq k$.
     A and B trivially satisfy (1) and (2).
  2. Both $\text{Depth}(A),\ \text{Depth}(B) \leq k$, and both $\text{Depth}(A\Theta),\ \text{Depth}(B\Theta) > k$.
     It follows from Theorem 3.11, L2, that $A\Theta \equiv B\Theta$.
  3. Both $\text{Depth}(A),\ \text{Depth}(B) \leq k$, and $\text{Depth}(A\Theta) > k$, but $\text{Depth}(B\Theta) \leq k$.
     It follows from Theorem 3.11, L2, that $A\Theta \sqsubseteq B\Theta$.
  4. $\text{Depth}(A) > k,\ \ \text{Depth}(B) \leq k$, and $\text{Depth}(A\Theta) > k,\ \text{Depth}(B\Theta) \leq k$.
     By induction $A \sqsubseteq B$. It follows from Theorem 3.11, L1, that $A\Theta \sqsubseteq B\Theta$.
  5. $\text{Depth}(A) > k,\ \text{Depth}(B) \leq k$, and $\text{Depth}(A\Theta),\ \text{Depth}(B\Theta) > k$.
     By induction $A \sqsubseteq B$. It follows from Theorem 3.11, L1, that $A\Theta \sqsubseteq B\Theta$. It follows from L2, that $B\Theta \sqsubseteq A\Theta$. Together this makes $A\Theta \equiv B\Theta$.
  6. $\text{Depth}(A), \text{Depth}(B) > k$, and $\text{Depth}(A\Theta),\ \text{Depth}(B\Theta) > k$.
     It follows from decomposing $\equiv$, twice applying Theorem 3.11, L1, that $A\Theta \equiv B\Theta$.

  It is easy to check that this ensures preservation of (1) and (2).

end of proof

This decomposition works because we use the order a posteriori. When the order $\prec$ is applied apriori, things go wrong:

Example 3.13 Let

$$C = \{p(0, s^3(0))\}, \{\neg\ p(X, s^3(0)), p(s(X), s^3(0))\}.$$

The $\prec$-order allows derivation of each $\{p(s^i(0), s^3(0))\}$ in the case of a priori application.

Now we are ready to prove the main theorem:

Theorem 3.14 Let C be a finite set of clauses in E^+. With a posteriori ordered resolution and factoring based on $\prec$ only a finite set of clauses can be derived from C.

proof:
It is sufficient to show that there exists a bound k on the Vardepth of derivable clauses. We take $k = \text{Depth}(C)$, the depth of a deepest clause in C.
We show that no clause with $\text{Vardepth}(C) > k$, can be introduced. First of all note that even unrestricted factoring cannot increase the variable depth of a clause, since by Theorem 2.4 the subsitution Θ must be a weak renaming on at least one of the literals, and because of this on the whole clause.
It remains to consider the resolution rule. We use the same decomposition as in Theorem 3.12. Let c be the first clause with $\text{Vardepth}(c) > k$. We show that there is no such c. Clause c can be seen as the propositional resolvent of $c_1 = \{A_1\Theta\} \cup R_1\Theta$ and $c_2 = \{\neg\, A_2\Theta\} \cup R_2\Theta$.
Let $B\Theta$ be a literal with $\text{Vardepth}(B\Theta) > k$. We show that $B\Theta$ cannot exist, due to the fact that its presence would block the derivation. First note that $\text{Vardepth}(A_i\Theta) \leq k$, because of Theorem 2.4. Because of this we have $\text{Vardepth}(A_i\Theta) < \text{Vardepth}(B\Theta)$.
Now suppose that $\text{Depth}(A_i\Theta) \geq \text{Depth}(B\Theta) > k$, then it would follow from Theorem 3.12 that $A_i\Theta \sqsubseteq B\Theta$, which is a contradiction with

$$\text{Vardepth}(A_i\Theta) < \text{Vardepth}(B\Theta) \text{ and } \text{Depth}(A_i\Theta) \geq \text{Depth}(B\Theta).$$

So it must be the case that $\text{Depth}(A_i\Theta) < \text{Depth}(B\Theta)$. But then the $\prec$-order would have blocked the derivation.
end of proof

4 Future Research

The obvious question is: Can the decidability results in ([Niv98]) be reproduced by a liftable, a posteriori order?

5 Acknowledgements

Part of the research was carried out during a visit to the Max Planck Institut in Saarbrücken, Germany.

References

[ANB96] H. Andréka, J. van Benthem, I. Németi, Modal Languages and Bounded Fragments of Predicate Logic, ILLC Research Report ML-96-03, 1996.

[BFL94] M. Baaz, C. Fermüller, A. Leitsch, A Non-Elementary Speed Up in Proof Length by Structural Clause Form Transformation, In LICS 94.

[BG90] L. Bachmair, H. Ganzinger, On restrictions of Ordered Paramodulation with Simplification, In CADE 10, pp. 427-441, 1990.

[BGG96] E. Börger, E. Grädel, Y. Gurevich, The Classical Decision Problem, Springer Verlag, Berlin Heidelberg, 1996.

[CL73] C-L. Chang, R. C-T. Lee, Symbolic Logic and Mechanical Theorem Proving, Academic Press, New York, 1973.

[DG79] B. Dreben, W.D. Goldfarb, The Decision Problem, Solvable Classes of Quantificational Formulas, Addision-Wesley Publishing Company, Inc. 1979.

[FLTZ93] C. Fermüller, A. Leitsch, T. Tammet, N. Zamov, Resolution Methods for the Decision Problem, Lecture Notes in Artificial Intelligence 679, Springer Verlag, 1993.

[Joyn76] W. H. Joyner, Resolution Strategies as Decision Procedures, J. ACM 23, 1 (July 1976), pp. 398-417, 1976.

[Leitsch97] A. Leitsch, The Resolution Calculus, Springer Verlag, 1997.

[McCune95] W. W. McCune, Otter 3.0 Reference Manual and Guide, Argonne National Laboratory, Mathematics and Computer Science Division, can be obtained from **ftp.mcs.anl.gov**, directory **pub/Otter**, 1995.

[Niv94a] H. de Nivelle, Application of Resolution Games to Resolution Decision Procedures, Internal Report 94-50, Department of Mathematics and Computer Science, Delft University of Technology, 1994.

[Niv94b] H. de Nivelle, Resolution Games and Non-Liftable Resolution Orderings, in CSL94, pp. 279-293, Springer Verlag, 1994.

[Niv95] H. de Nivelle, Ordering Refinements of Resolution, Ph. D. Thesis, Delft University of Technology, 1995.

[Niv98] H. de Nivelle, Resolution Decides the Guarded Fragment, Internal Report, to appear, 1998.

[Robins65] J. A. Robinson, A Machine Oriented Logic Based on the Resolution Principle, Journal of the ACM, Vol. 12. pp. 23-41, 1965

[Tam90] T. Tammet, The Resolution Program, Able to Decide some Solvable Classes, in COLOG-88, Springer LNCS, pp. 300-312, 1990.

[Weid96] C. Weidenbach, (Max-Planck-Institut für Informatik), The Spass & Flotter Users Guide, Version 0.55, can be obtained from **ftp.mpi-sb.mpg.de**, directory **pub/SPASS**, 1997.

[Zam72] N.K. Zamov, On a Bound for the Complexity of Terms in the Resolution Method, Trudy Mat. Inst. Steklov 121, pp. 1-10, 1972.