Chapter 3

SEMANTIC CLASH RESOLUTION AS DECISION PROCEDURE

3.1 INTRODUCTION AND DEFINITIONS

Semantic resolution was one of the first resolution refinements developed after Robinson's famous paper on resolution [Rob65a]. Already in the same year, this author proposed the refinement of hyperresolution [Rob65b]; even somewhat earlier S. Maslov [Mas64] used a clash method defined within sequent calculus to decide the class $\exists^*\forall^*\exists^*$Krom. The essential feature of clash methods is the possibility of "macro-inference", that is performing more than one resolution step at once and forgetting the intermediate results. Still hyperresolution is one of the most efficient resolution refinements. In [Sla67] Slagle generalized Robinson's idea to what is generally known as semantic clash resolution (RSC). The basic idea is to choose a Herbrand model $\mathfrak{M}$ and separate true clauses (in $\mathfrak{M}$) from false clauses (in $\mathfrak{M}$); true clauses may only be resolved with false clauses. Even with the clash property and with restricted factoring, the method is complete and can be combined with the deletion strategy (tautology-elimination + subsumption).

In this chapter we show that semantic clash resolution is well suited to decide a wide range of clause classes. That means we can give some general syntactic criteria on clauses which guarantee termination of RSC for an appropriate setting (which can be constructed effectively). Particularly we generalize some results on decidable Horn classes, which were based on hyperresolution and unit resolution ([Fer90], [Lei90]). The resolution refinements used in this chapter do not only decide interesting clause classes, but at the same time yield quite efficient theorem provers (because they are complete). We also show that the decidable classes are quite "sharp" in the sense that slight weakening of restrictions already gives undecidable classes.

We now give some basic definitions.

<u>DEFINITION</u> 3.1.1.: Let $\mathcal{C}$ be a set of clauses and $\{P_1, \dots, P_n\}$ the set of predicate symbols occuring in $\mathcal{C}$. A <u>setting</u> $\mathfrak{m}$ is a Herbrand interpretation which, for every P_i, assigns all ground atoms $P_i(\bar{t})$ to true or all $P_i(\bar{t})$ to false ($\bar{t}$ is a ground term vector of appropriate arity).

<u>Remark:</u> The definition above is not the most general one, but suffices for our purposes; thus we follow the definitions of Chang & Lee ([CL73]) rather than that of Loveland ([Lov78]).

<u>EXAMPLE</u> 3.1.1.:

$\mathcal{C} = \{\{P(b)\}, \{\neg P(x), P(f(x))\}, \{\neg P(f(f(a)))\}\}$; $H = \{a, b, f(a), (f(b), \dots\}$

$\mathfrak{m}_1 = \{P(t) \ / \ t \in H\}$, $\mathfrak{m}_2 = \{\neg P(t) \ / \ t \in H\}$ are the only two possible settings. $\mathfrak{m} = \{\neg P(a), P(b), \neg P(f(a)), P(f(b)), \neg P(f^2(a)), \dots\}$ is a Herbrand interpretation, but is not allowed as a setting. With respect to $\mathfrak{m}_1$, $\{P(b)\}$ and $\{\neg P(x), P(f(x))\}$ are true in $\mathfrak{m}_1$ (every ground instance is true in $\mathfrak{m}_1$), but $\{\neg P(f^{(2)}(a))\}$ is false $\mathfrak{m}_1$.

<u>DEFINITION</u> 3.1.2.: Let $\mathcal{C}$ be a set of clauses and $\mathfrak{m}$ be a setting for $\mathcal{C}$. A resolvent of two clauses C_1, C_2 in $\mathcal{C}$ is called an <u>$\mathfrak{m}$ - resolvent</u> if one of C_1, C_2 is false in $\mathfrak{m}$ (note that, by definition of a setting, no resolvent exists if both clauses are false in $\mathfrak{m}$).

For the set of clauses $\mathcal{C}$ in example 3.1.1. there is only one possible $\mathfrak{m}_1$ - resolvent; this one is $\{\neg Pf(a))\}$ (resolve the second with the third clause). After this resolution we get

$$\mathcal{C}_1 = \{ \underbrace{\{P(b)\}}_{C_1}, \underbrace{\{\neg P(x), P(f(x))\}}_{C_2}, \underbrace{\{\neg P(f(a))\}}_{C_3}, \underbrace{\{\neg P(f^2(a))\}}_{C_4} \}$$

While resolvents from (C_2, C_3) and (C_2, C_4) are allowed now, the only new one is $\{\neg P(a)\}$ (derived from (C_2, C_3)) and we get $\mathcal{C}_2 = \{C_1, C_2, C_3, C_4, \{\neg P(a)\}\}$. It is easy to verify that no new resolvents are derivable from $\mathcal{C}_2$. But $\mathcal{C}_2$ does not contain $\square$; because $\mathfrak{m}$ - resolution is complete (a consequence of the completeness of RSC [CL73]) we have proved that $\mathcal{C}$ is satisfiable. Starting with setting $\mathfrak{m}_2$ we would not do so well anymore; in this case we could derive the sequence $\{P(b)\}$, $\{P(f(b))\}$, $\{P(f^2(b))\}$, ... and resolution would never stop on $\mathcal{C}$. While $\mathfrak{m}_1$- and $\mathfrak{m}_2$ - resolution are both complete, $\mathfrak{m}_1$ - resolution decides $\mathcal{C}$ (that means it is terminating on $\mathcal{C}$), but $\mathfrak{m}_2$- resolution does not. Later we will show that for some classes appropriate settings (which yield termination) can be found automatically by an analysis of the term- and variable structure in clauses.

<u>DEFINITION</u> 3.1.3.: An $\mathfrak{M}$ - resolvent of clauses C_1, C_2 s.t. C_1 is true in $\mathfrak{M}$ and C_2 is false in $\mathfrak{M}$, is said to be defined "<u>under restricted factoring</u>" from C_1, C_2 if factoring substitutions are only allowed in C_2.

<u>EXAMPLE</u> 3.1.2.:

$$\mathcal{C} = \{\underbrace{\{P(x), P(y)\}}_{C_1}, \underbrace{\{\neg P(u), \neg P(v)\}}_{C_2}\}, \quad \mathfrak{M} = \{P(a)\}.$$

Let $A = \{P(y)\}$, $B = \{\neg P(u), \neg P(v)\}$ and $\lambda = \{y/u, y/v\}$.
Then $\{P(x)\} = (C_1 - A)\lambda \cup (C_2 - B)\lambda$ is defined under restricted factoring.
If, on the other hand, we set $A = \{P(x), P(y)\}$, $B = \{\neg P(u)\}$ and $\lambda = \{u/x, u/y\}$ then $(C_1 - A)\lambda \cup (C_2 - B)\lambda$ $(= \{\neg P(v)\})$ is not defined under restricted factoring.

<u>DEFINITION</u> 3.1.4.: A <u>semantic clash sequence</u> is a sequence Γ of the form $(C; D_1, \dots, D_n)$ where $C, D_1, \dots, D_n$ are clauses in some set of clauses $\mathcal{C}$, C is true and all D_i are false in a setting $\mathfrak{M}$. C is called nucleus, the D_i electrons of Γ.
Let $R_0 = C$ and

R_{i+1} = an $\mathfrak{M}$ - resolvent of R_i and D_{i+1} defined under restricted factoring (if it exists).

If R_n (defined after a linear deduction of n resolution steps) is false in $\mathfrak{M}$ (it might be $\square$) then R_n is called semantic clash resolvent of Γ with respect to $\mathfrak{M}$ ($SC\mathfrak{M}$ - resolvent).

<u>EXAMPLE</u> 3.1.3.:

$$\text{Let } \mathcal{C} = \{\underbrace{\{P(x), Q(x,y), \neg Q(x,g(y)), \neg Q(x,f(y))\}}_{C_1}, \underbrace{\{Q(u,v), R(v)\}}_{C_2}, \underbrace{\{Q(u',g(u')), R(g(u'))\}}_{C_3}\}$$

$\Gamma = (C_1; C_2, C_3)$ is a clash sequence w.r.t. $\mathfrak{M} = \{\neg P(s), \neg Q(s,t), \neg R(s) \mid s,t \in H(\mathcal{C})\}$.

$R_0 = C_1$

$R_1 = \{P(x),\ Q(x,y),\ R(f(y)),\ \neg Q(x,g(y))\}$

$R_2 = \{P(x), Q(x,y),\ R(f(y)),\ R(g(y))\}$

R_2 is false in $\mathfrak{m}$ and thus is a SC$\mathfrak{m}$ - resolvent of Γ.

Note that there are two possibilities to define a resolvent R_1; but by resolving the literals $\neg Q(x,g(y))$, $Q(u,v)$ we get $R_1' = \{P(x), Q(x,y), R(g(y)), \neg Q(x,f(y))\}$.

But R_1' cannot be resolved with C_3. We get that Γ defines only one clash resolvent in this case; in general there are several possibilities to define the R_i and there are several resolvents of a clash.

While the completeness of semantic clash resolution with literal ordering in the electrons is contained in the standard literature on automatic theorem proving [CL 73], [Lov 78], the completeness of RSC$\mathfrak{m}$ with restricted factoring (as in definition 3.1.3.) was shown by H. Noll in 1980 [Nol 80]. In this chapter we need a further refinement to decide some clause classes; in order to control the growth of the clash resolvents we have to keep all resolvents in condensed form. For definition and properties of condensing see chapter 2.

<u>DEFINITION</u> 3.1.5.: A semantic clash $\Gamma = (C; D_1,\dots,D_n)$ is called <u>condensed</u> if C and all clauses D_i are condensed.

Like in other papers on resolution decision procedures we don't look at the deductions themselves, but rather at the set of clauses derivable (by some refinement) from the set $\mathcal{C}$.

<u>DEFINITION</u> 3.1.6.: Let $\mathcal{C}$ be a set of clauses and $\mathfrak{m}$ be a setting for $\mathcal{C}$. We define the operators RSC$\mathfrak{m}$D, $R^i_{\mathfrak{m}}$, $R^*_{\mathfrak{m}}$ as follows: RSC$\mathfrak{m}$D($\mathcal{C}$) is the set of all clash resolvents definable by clashes from $\mathcal{C}$ which are represented in condensed form.

More formally:

RSC$\mathfrak{m}$D($\mathcal{C}$) = the set of all C_{cond} s.t. C is a SC$\mathfrak{m}$ - resolvent of a semantic clash in $\mathcal{C}$ according to definition 3.1.4. under equivalence relation $\sim_v$, defined as C $\sim_v$ D iff C is a variant of D; clearly $\sim_v$ is an equivalence relation on the set of clauses.

Furthermore we define recursively:

$\mathcal{C}_0 = \mathcal{C}_{cond} = \{C \,/\, C \in \mathcal{C}, C \text{ true in } \mathfrak{m}\} \cup \{C_{cond} \,/\, C \in \mathcal{C}, C \text{ false in } \mathfrak{m}\}$.

$\mathcal{C}_{i+1} = (\mathcal{C}_i \cup RSC\mathfrak{m}D(\mathcal{C}_i))/_{\sim_v}$ and $R^i_{\mathfrak{m}}(\mathcal{C}) = \mathcal{C}_i$, $\quad R^*_{\mathfrak{m}}(\mathcal{C}) = \bigcup_{i \in N} R^i_{\mathfrak{m}}(\mathcal{C})$.

By definition, $R^*_{\mathfrak{m}}(\mathcal{C})$ is the set of all clauses derivable by RSC$\mathfrak{m}$D from $\mathcal{C}$. Correctness of SC$\mathfrak{m}$D - resolution means that $R^*_{\mathfrak{m}}(\mathcal{C})$ contains $\square$ only if $\mathcal{C}$ is unsatisfiable; completeness means that the unsatisfiability of $\mathcal{C}$ implies $\square \in R^*_{\mathfrak{m}}(\mathcal{C})$ (for every set of clauses $\mathcal{C}$ and every setting $\mathfrak{m}$ for $\mathcal{C}$).

We now prove the completeness of RSC$\mathfrak{m}$D on the basis of the completeness of RSC$\mathfrak{m}$ (which follows from [Nol80]).

<u>THEOREM</u> 3.1.1.: SC$\mathfrak{m}$D - resolution is complete. More formally:

Let $\mathcal{C}$ be an unsatisfiable set of clauses and $\mathfrak{m}$ be a setting for $\mathcal{C}$; then $\square \in R^*_{\mathfrak{m}}(\mathcal{C})$.

<u>Proof</u>:

Let $S^*_{\mathfrak{m}}(\mathcal{C})$ be the set of clauses deducible from $\mathcal{C}$ by RSC$\mathfrak{m}$ without condensing ($S^*_{\mathfrak{m}}$ can be defined formally like in definition 3.1.6 omitting the condensation operator).

We show that for every $C \in S^*_{\mathfrak{m}}(\mathcal{C})$ there is a $C' \in R^*_{\mathfrak{m}}(\mathcal{C})$ s.t. $C' \leq_s C$. Because $\square$ is subsumed only by $\square$ itself, $\square \in S^*_{\mathfrak{m}}(\mathcal{C})$ immediately implies $\square \in R^*_{\mathfrak{m}}(\mathcal{C})$ and the completeness of RSC$\mathfrak{m}$D follows from that of RSC$\mathfrak{m}$.

We proceed by induction on i for $S^i_{\mathfrak{m}}(\mathcal{C})$.

$i = 0$: $\quad S^i_{\mathfrak{m}}(\mathcal{C}) = \mathcal{C}$. $R^i_{\mathfrak{m}}(\mathcal{C}) = \mathcal{C}_{cond}$.

By definition of the condensation operator we have $C_{cond} \leq_s C$ for all C and thus for every $C \in S^i_{\mathfrak{m}}(\mathcal{C})$ there is a $C' \in R^i_{\mathfrak{m}}(\mathcal{C})$ with $C' \leq_s C$ for $i = 0$.

(IH) Suppose that for every $C \in S^i_{\mathfrak{m}}(\mathcal{C})$ there is a $C' \in R^i_{\mathfrak{m}}(\mathcal{C})$ s.t. $C' \leq_s C$.

<u>Case i + 1:</u>

Let $C \in S^{i+1}_{\mathfrak{m}}(\mathcal{C}) - S^i_{\mathfrak{m}}(\mathcal{C})$. Then C is resolvent of a clash $\Gamma = (E; D_1,\dots,D_n)$. That means $C = R_n$ for R_j defined as in definition 3.1.4. By (IH) there are clauses D_j' for $j = 1,\dots,n$ s.t. $D_j' \in R^i_{\mathfrak{m}}(\mathcal{C})$ and $D_j' \leq_s D_j$ and E' s.t. $E' \leq_s E$. We show now that either $D_j' \leq_s C$ for some $j \in \{1,\dots,n\}$ or there is a clash resolvent C' out of $\{E', D_1', .., D_n'\}$ s.t. $C' \leq_s C$; the corresponding clash Γ' may be shorter than Γ.

Again, we use induction on j.

$R_0(\Gamma) = E$, $R_0' = E'$ and $E' \leq_s E$ by (IH).

(IH*) Suppose that either $R_k' \leq_s R_j(\Gamma)$ or $D_k \leq_s C$ for some $k \leq j$.

The case for $j = 0$ is obviously valid.

Case $j + 1$:

a) $D_k \leq_s C$ for some $k \leq j$.

b) $R_k' \leq_s R_j(\Gamma)$.

In case a) (IH*) is trivial for j+1, because then $D_k \leq_s C$ for some $k \leq j+1$.

b) For simplicity we write $R_j = R_j(\Gamma)$.

Let $R_{j+1} = (R_j - \{L\})\sigma \cup (D_{j+1} - A)\sigma$. s.t. σ is m.g.u. of $\{L^d\} \cup A$ (note that factoring may only occur in D_{j+1} and thus $\{L\}$ must be a singleton set).

By (IH) we have $D_{j+1}' \leq_s D_{j+1}$ and by (IH*) $R_k' \leq_s R_j$.

Suppose that either $R_k' \leq_s R_j - \{L\}$ or $D_{j+1}' \leq_s D_{j+1} - A$;

If $R_k' \leq_s R_j - \{L\}$ then, by $R_j - \{L\} \leq_s R_{j+1}$ we get $R_k' \leq R_{j+1}$.

If $D_{j+1}' \leq_s D_{j+1} - A$ then $D_{j+1}' \leq_s R_l$ for every $l \geq j+1$.

This holds, because all literals in $(D_{j+1} - A)\sigma$ are false in $\mathfrak{m}$ and thus cannot be resolved away by further resolutions in the clash. Thus, in R_n there must be a subclause of the form $(D_{j+1} - A)\sigma\vartheta$, where ϑ is a substitution composed from the m.g.u.'s of the further resolvents in the clash.

It remains to settle the case where $R_k' \not\leq_s R_j - \{L\}$ and $D_{j+1}' \not\leq_s D_{j+1} - A$. Then there is a substitution η s.t. $R_k'\eta \subseteq R_j$ and $B\eta = \{L\}$ for a set $B \subseteq R_k'$. Moreover, there is a substitution ϑ and a set $F \subseteq D_{j+1}'$ s.t. $F\vartheta \subseteq A$, $D_{j+1}'\vartheta \subseteq D_{j+1}$. Furthermore suppose that B and F are maximal, that means $(R_k' - B)\vartheta \cap \{L\} = \emptyset$. Then (supposing that D_{j+1}' and R_k' do not share variables) $B \cup F^d$ is unifiable by a m.g.u. σ'; by $F \leq_s A$, $B \leq_s \{L\}$ it must hold $\sigma' \leq_s \sigma$. Thus we define $R_{k+1}' = (R_k' - B)\sigma' \cup (D_{j+1}' - F)\sigma'$; clearly R_{k+1}' is an $\mathfrak{m}$ - resolvent of R_k' and D_{j+1}', but not (in general) under restricted factoring.

$$\text{Thus } R_{j+1} = (R_j - \{L\})\sigma \cup (D_{j+1} - A)\sigma = ((R_j - \{L\}) \cup (D_{j+1} - A))\sigma =$$
$$((R_k' - B)\eta \cup (D_{j+1}' - F)\vartheta)\sigma = ((R_k' - B) \cup (D_{j+1}' - F))\mu\sigma$$

for some substitution μ.

But $R_{k+1}' = ((R_k' - B) \cup (D_{j+1}' - F))\sigma'$ where σ' is a m.g.u. of $B \cup F^d$. Thus we get $R_{k+1}' \leq_s R_{j+1}$.

Concluding the induction proof (*) we get that there is a clash resolvent C' out of $\{E', D_1', .., D_n'\}$ without restriction of factoring s.t. $C' \leq_s C$ or $D_j' \leq_s C$ for some j.

By lemma 3.1.1. (to be proved afterwards) the clash resolvent C' can be obtained from a clash out of $\{E', D_1', \ldots, D_n'\}$ under restricted factoring (the clash under restriction of factoring may then be longer than the original one). More exactly, we derive from lemma 3.1.1. that there is a

$C'' \in RSC\mathfrak{m}(\{E', D_1', \ldots, D_n'\})$ s.t. C' is a factor of C''.

But then $C'' \leq_s C' \leq_s C$ and C''_{cond}, which is in $RSC\mathfrak{m}D(\{E', D_1', \ldots, D_n'\})$, subsumes C by $C'' =_s C''_{cond}$.

We conclude that $C''_{cond} \in R^{i+1}_{\mathfrak{m}}(\mathcal{C})$ and $C''_{cond} \leq_s C$.

In any case we have settled "i+1" of the main induction proof and thus for every $E \in S^{i+1}_{\mathfrak{m}}(\mathcal{C})$ there is a $E' \in R^{i+1}_{\mathfrak{m}}(\mathcal{C})$ s.t. $E' \leq_s E$.

We conclude that for every $E \in S^{*}_{\mathfrak{m}}(\mathcal{C})$ there is an $E' \in R^{*}_{\mathfrak{m}}(\mathcal{C})$ s.t. $E' \leq_s E$.

Q.E.D.

It is easily verified that the proof of theorem 3.1.1. can be modified to a proof of completeness for $SC\mathfrak{m}D$ - resolution with forward subsumption.

Let $RSC\mathfrak{m}$ the same operator as $RSC\mathfrak{m}D$, but without condensing the resolvents. The following technical lemma was necessary to prove theorem 3.1.1.

<u>LEMMA</u> 3.1.1.: Let $\mathcal{C}$ be a set of clauses and $\mathfrak{m}$ be a setting for $\mathcal{C}$. Then for every $C \in RSC\mathfrak{m}(\mathcal{C})$ without restriction of factoring there is a $C' \in RSC\mathfrak{m}(\mathcal{C})$ s.t. $C' \leq_s C$.

<u>Proof:</u>

By the basic lemma in [Nol 80], for every resolvent C of clauses D, E there is a resolution deduction out of $\{D, E\}$ with "half-factoring" giving a clause C' s.t. $C'\eta = C$ for some instantiation η. If D is true and E is false in $\mathfrak{m}$, then by restricting factoring to negative clauses, we get in fact a semantic clash resolvent C' under restricted factoring. In theorem 2 [Nol 80] it is shown that this property can be extended to whole clashes in semantic resolution. Thus for every $C \in RSC\mathfrak{M}(\mathcal{C})$ without restriction of factoring there is a $C' \in RSC\mathfrak{M}(\mathcal{C})$ (with restriction of factoring) s.t. $C'\eta = C$ for some substitution η. Particularly, we have $C' \leq_s C$.

Q.E.D.

Parts of the proof of theorem 3.1.1. can be found in the proofs of subsumption results in [Lov 78]. However, Loveland uses another concept of resolution than we do here. Because there might occur some dangerous side effects due to factoring (see [Lei 89]), we have represented these technical matters in more detail. We also note, that H. Noll [Nol 80] uses Robinson's concept on which also this work is based.
We conclude the chapter with some useful notations and definitions.

DEFINITION 3.1.7.: Let $\mathcal{C}$ be a set of clauses and $\{P_1, \dots, P_n\}$ the set of predicate symbols in $\mathcal{C}$. Let $\mathfrak{M}p = \{P_1(\bar{s}_1), \dots, P_n(\bar{s}_n) / \bar{s}_i$ are $H(\mathcal{C})$ - tupels of appropriate arity$\}$; then $\mathfrak{M}p$ is called the positive setting for $\mathcal{C}$. Similarly $\mathfrak{M}_n = \{\neg P_1(\bar{s}_1), \dots, \neg P_n(\bar{s}_n) / \bar{s}_i$ are $H(\mathcal{C})$ - tupels of appropriate arity$\}$ is called the negative setting for $\mathcal{C}$.

Let $\mathfrak{M}$ be a setting for $\mathcal{C}$ and $C \in \mathcal{C}$. Then C_{neg} is the maximal subset of C which is false in $\mathfrak{M}$, $C_{pos} = C - C_{neg}$. For clauses C which are false in $\mathfrak{M}$ we clearly have $C = C_{neg}$.
If $\mathcal{C}$ is a clause set then, under negative setting $C_+ = C_{neg}$, under positive setting $C_- = C_{neg}$. Thus C_+, C_- denote the syntactical positivity and negativity, while C_{neg}, C_{pos} describes the semantical status w.r.t. a setting.

3.2 GENERAL CLAUSE CLASSES

In [Lei 90], [Fer90] hyperresolution has been used as decision procedure for some Horn classes. In this chapter we will present the results obtained in these papers and show that, for many of the classes, the Horn restriction is not essential. Moreover, instead of using different forms of hyperresolution (positive and negative) we define a single resolution refinement, which - under appropriate choice of the setting (to be done algorithmically) - decides all classes in [Lei 90], [Fer 90].

We begin with the Horn classes KI, KII, defined in [Lei 90]. For definitions of Horn clauses, rule, fact, goal see chapter 2.

KI: A set of clauses $\mathcal{C}$ belongs to KI if it holds:

a) $\mathcal{C}$ is Horn

b) If $C = \{P, \neg P_1, \dots, \neg P_n\}$ is a rule clause in $\mathcal{C}$ then $\tau(P) = 0$ and $V(P) \subseteq V(\{\neg P_1, \dots, \neg P_n\})$

c) Facts are ground.

It is easy to see that DATALOG [CGT 90] is a subclass of KI (for DATALOG we also have $\tau(C) = 0$ for all $C \in \mathcal{C}$).
It was proved in [Lei90] that positive hyperresolution always terminates on all $\mathcal{C} \in$ KI and thus gives a decision procedure for KI. Because the fixed ordering of the rule clauses was not used in the proof, it is immediately verifiable that RSC$\mathfrak{m}$D decides KI for negative setting $\mathfrak{m}$ (all ground atoms are set to false).

The second class was KII:

A set of clauses $\mathcal{C}$ belongs to KII if it holds:

a) $\mathcal{C}$ is Horn

b) If $C = \{P, \neg P_1, \dots, \neg P_n\}$ is a rule clause then $\tau(\{\neg P_1, \dots, \neg P_n\}) = 0$ and $V(\{\neg P_1, \dots, \neg P_n\}) \subseteq V(\{P\})$

c) Goals are ground.

It is quite obvious that, in some sense, KII is a mirror image of KI. This feature also applies to positive hyperresolution: While for KI the depth of clash resolvents was monotonically decreasing, it is increasing for KII. Thus RSC$\mathfrak{m}$D does not terminate on KII for negative setting $\mathfrak{m}$; however it was shown, that production of resolvents deeper than $\tau(\mathcal{C})$ is useless and that positive hyperresolution can be turned into a decision algorithm by cutting the term depth at $\tau(\mathcal{C})$. On the other hand , RSC$\mathfrak{m}$D for positive setting $\mathfrak{m}$ always terminates an KII on thus yields a decision procedure. Both KI, KII share some common syntax property which is formulated in the following class:

PVD (positive - variable dominated).

DEFINITION 3.2.1: A set of clauses $\mathcal{C}$ belongs to PVD if it holds:
There exists a setting $\mathfrak{m}$ for $\mathcal{C}$ s.t.

PVD 1) All clauses in $\mathcal{C}$ which are false in $\mathfrak{m}$ are ground.

PVD 2) If C is in $\mathcal{C}$ and C is true in $\mathfrak{m}$ then for $x \in V(C_{neg})$ it holds

$$\tau_{max}(x, C_{neg}) \leq \tau_{max}(x, C_{pos}) \text{ and } V(C_{neg}) \subseteq V(C_{pos}).$$

Note that there is no restriction on clauses containing only literals true in $\mathfrak{m}$. It is easy to see that KI $\cup$ KII $\subseteq$ PVD. For KI we have to choose negative, for KII positive setting. Even if restricted to Horn clauses, KI $\cup$ KII is a proper subset of PVD ($\tau(C_{neg}) = 0$ implies $\tau_{max}(x, C_{neg}) \leq \tau_{max}(x, C_{pos})$ for $x \in C_{neg}$, but - in PVD- $\tau(C_{neg})$ need not be zero).

The term "there exists a setting $\mathfrak{m}$" in definition 3.2.1. could suggest, that although there might be such an $\mathfrak{m}$ we cannot find it. Fortunately this is not the case because, by our definition of setting, there are only 2^n settings, n being the number of different predicate symbols in a set of clauses $\mathcal{C}$. Moreover, after having decided for a particular $\mathfrak{m}$, PVD 1) and PVD 2) can be decided algorithmically. We do not deal with algorithms deciding "$\mathcal{C} \in$ PVD" here, but finding efficient ones is certainly an interesting task.

EXAMPLE 3.2.1.:

$$\mathcal{C} = \{\underbrace{\{P(x), Q(g(x,x))\}}_{C_1}, \underbrace{\{\neg Q(y), R(x,y)\}}_{C_2}, \underbrace{\{\neg R(a,a), \neg R(f(b),a)\}}_{C_3}$$

$$\underbrace{\{R(f(x), y)\}}_{C_4}, \underbrace{\{\neg P(x), \neg P(f(x))\}}_{C_5}\}$$

$\mathcal{C}$ is not Horn and cannot be transformed into Horn by changing the signs of the literals.

The following setting $\mathfrak{m} = \{Q(s), R(s,t), \neg P(s) \mid s,t \in H(\mathcal{C})\}$ "brings" $\mathcal{C}$ into PVD. Indeed,

$(C_1)_{neg} = \{P(x)\}$, $(C_1)_{pos} = \{Q(g(x,x))\}$.

$(C_2)_{neg} = \{\neg Q(y)\}$, $(C_2)_{pos} = \{R(x,y)\}$

$C_3 = (C_3)_{neg} = \{\neg R(a,a), \neg R(f(b),a)\}$

$C_4 = (C_4)_{pos} = \{R(f(x),y)\}$

$C_5 = (C_5)_{pos} = \{\neg P(x), \neg P(f(x))\}$

and PVD 1), PVD 2) are fulfilled for all clauses.

<u>THEOREM</u> 3.2.1.: **SC$\mathfrak{m}$D - resolution is a decision procedure for PVD.**
More precisely:

There is an algorithm defining for every $\mathcal{C} \in$ PVD a setting $\mathfrak{m}$ s.t. $R^{*}_{\mathfrak{m}}(\mathcal{C})$ is finite.

<u>Proof:</u>

Let $\mathcal{C} \in$ PVD; because there are only finitely many settings for $\mathfrak{m}$ and PVD 1), PVD 2) can be decided for every $\mathfrak{m}$, we eventually find the setting $\mathfrak{m}$ fulfilling PVD 1), PVD 2).

We show now that $R^{*}_{\mathfrak{m}}(\mathcal{C})$ is finite.

For this purpose it is enough to show both a) and b):

a) $R^{*}_{\mathfrak{m}}(\mathcal{C}) - \mathcal{C}$ contains ground clauses only

b) For all $E \in R^{*}_{\mathfrak{m}}(\mathcal{C}) - \mathcal{C}$ it holds $\tau(E) \leq d$, where

$d = \max\{\tau(C_{neg}) \,/\, C \in \mathcal{C}\}$.

Because there are only finitely many ground clauses of term depth $\leq d$ over the Herbrand universe of $\mathcal{C}$, a), b) guarantee that $R^{*}_{\mathfrak{m}}(\mathcal{C})$ is indeed finite.

We prove a) and b) for $R^{i}_{\mathfrak{m}}(\mathcal{C})$ using induction on i.

$i = 0$: $R^{0}_{\mathfrak{m}}(\mathcal{C}) - \mathcal{C} = \emptyset$ and a) is trivially true. The same holds for b).

(IH) Suppose that a - i) $R^{i}_{\mathfrak{m}}(\mathcal{C}) - \mathcal{C}$ contains only ground clauses; and

b - i) For all $E \in R^{i}_{\mathfrak{m}}(\mathcal{C}) - \mathcal{C}$ we have $\tau(E) \leq d$;

holds.

a-i+1), b-i+1):

Let $\Gamma = (C; D_1, \dots, D_n)$ be a semantic $\mathfrak{m}$ - clash defined by clauses in $R^{i}_{\mathfrak{m}}(\mathcal{C})$. Because all false clauses in $\mathcal{C}$ are ground by PVD 1) and because, by (IH) all derived (false) clauses are ground, the D_i must be ground.

Let $R_0 = C$

R_{i+1} = an $\mathfrak{m}$ resolvent of R_i and D_{i+1} according to definition 3.1.4.

If Γ is resolvable then there is a clash resolvent R_n.

For every pair (R_i, D_{i+1}) and $0 \leq i < n$ there is a literal $L_i \in D_{i+1}$ and a literal $M_i \in R_i$ s.t. the m.g.u. λ_i of the resolution unifies $\{M_i, L_i^d\}$. Note that no factoring substitution is required in the resolution because D_{i+1} is ground. Moreover all the λ_i are ground substitutions fulfilling $M_i\lambda_i = L_i^d$.

Let $\lambda = \lambda_1 \dots \lambda_n$; because all λ_i are ground, we have $M_i \lambda = M_i \lambda_i = L_i^d$.
Because R_n must be false in $\mathfrak{m}$, $C_{pos} = \{M_0, \dots, M_{n-1}\}$ and

$C_{pos} \lambda \subseteq \{L_0^d, \dots, L_{n-1}^d\}$, $\quad R_n = C_{neg}\lambda \cup \bigcup_{i=0}^{n-1} (D_{i+1} - \{L_i\})$.

By PVD 2) we have $V(C_{neg}) \subseteq V(C_{pos})$ (note that C, as every true clause in $R^*_{\mathfrak{m}}$, must be in $\mathcal{C}$).
Because $C_{pos} \lambda$ is ground, $C_{neg}\lambda$ must be ground too.
Because also $\bigcup^{n-1} (D_{i+1} - \{L_i\})$ is ground, also R_n is ground and a- i + 1) is shown.

By PVD 2) we also have $\tau_{max}(x, C_{neg}) \leq \tau_{max}(x, C_{pos})$ for all $x \in V(C_{neg})$.
Either $\tau(C_{neg}\lambda) = \tau(C_{neg})$ or there is a $x \in V(C_{neg})$
s.t. $\tau(C_{neg}\lambda) = \tau_{max}(x, K) + \tau(x\lambda)$ for some $K \in C_{neg}$.
If $\tau(C_{neg}\lambda) = \tau(C_{neg})$, then by definition of d and by (IH)-b-i) we have $\tau(R_n) \leq d$.

If $\tau(C_{neg}\lambda) = \tau_{max}(x, K) + \tau(x\lambda)$ then we know that
$\tau_{max}(x, C_{neg}) \leq \tau_{max}(x, C_{pos})$ and thus
$\tau_{max}(x, K) + \tau(x\lambda) \leq \tau_{max}(x, C_{pos}) + \tau(x\lambda) \leq \tau(C_{pos}\lambda)$.
It follows $\tau(C_{neg}\lambda) \leq \tau(C_{pos}\lambda)$.

But
$C_{pos} \lambda \subseteq \{L_0^d, \dots, L_{n-1}^d\}$ for $L_i \in D_{i+1}$ and thus
$\tau(C_{pos}\lambda) \leq \max\{\tau(D_i) \,/\, i = 1, \dots, n\}$.
Because, by (IH)-b-i) $\tau(D_i) \leq d$ for all i = 1, ... ,n we also get
$\tau(C_{pos}\lambda) \leq d$ and therefore $\tau(C_{neg}\lambda) \leq d$.
$\tau(R_n) \leq \max\{\tau(C_{neg}\lambda), \tau(\bigcup_{i=0}^{n-1}(D_{i+1} - \{L_i\}))\} \leq d$
is an immediate consequence.

This concludes the proof of b-i+1).

By the induction rule we get for all $E \in R^*_{\mathfrak{m}}(\mathcal{C})$:

a) E is ground
b) $\tau(E) \leq d$.

Q.E.D.

EXAMPLE 3.2.2.:

$\mathcal{C} = \{\{P(x), Q(g(x,x))\}, \{\neg Q(y), R(x,y)\}, \{\neg R(a,a), \neg R(f(b),a)\}, \{R(f(x),y)\},$
$\{P(x), \neg P(f(x))\}, \{\neg P(a)\}\}.$

We apply RSC$\mathfrak{m}$D to decide $\mathcal{C}$.

First we realize that $\mathcal{C}$ is indeed in PVD; we only have to find the setting $\mathfrak{m} = \{Q(s), R(s,t), \neg P(s) \ / \ s,t \in H(\mathcal{C})\}$.

The next step is the computation of $R^{*}_{\mathfrak{m}}(\mathcal{C})$.

$R^{0}_{\mathfrak{m}}(\mathcal{C}) = \mathcal{C}.$

$R^{1}_{\mathfrak{m}}(\mathcal{C}) = \mathcal{C} \cup \{\{\neg R(a,a)\}, \{\neg Q(a), \neg R(f(b),a)\}, \{\neg Q(a), \neg R(a,a)\}\}$

$R^{2}_{\mathfrak{m}}(\mathcal{C}) = R^{1}_{\mathfrak{m}}(\mathcal{C}) \cup \{\{\neg Q(a)\}\}$

$R^{3}_{\mathfrak{m}}(\mathcal{C}) = R^{2}_{\mathfrak{m}}(\mathcal{C})$ and thus $R^{*}_{\mathfrak{m}}(\mathcal{C}) = R^{2}_{\mathfrak{m}}(\mathcal{C}).$

Because $\Box \notin R^{*}_{\mathfrak{m}}(\mathcal{C})$ we know that $\mathcal{C}$ is satisfiable.

Note that choosing an arbitrary setting on $\mathcal{C}$ without syntax analysis of clauses in $\mathcal{C}$ can lead to non-termination. If we take for example $\mathfrak{m}_P$ we can derive $\{\neg P(f^{(i)}(a))\}$ for all $i \geq 0$ and thus $R_{\mathfrak{m}_P}{}^{*}(\mathcal{C})$ is infinite.

By computing $R^{*}_{\mathfrak{m}}(\mathcal{C})$ we do not only know that $\mathcal{C}$ is satisfiable, we are also able to construct a Herbrand model for $\mathcal{C}$.

$R^{*}_{\mathfrak{m}}(\mathcal{C}) = \mathcal{C} \cup \{\{\neg R(a,a)\}, \{\neg Q(a)\}, \{\neg Q(a), \neg R(f(b),a)\}, \{\neg Q(a), \neg R(a,a)\}\}.$

If we perform subsumption in the set of the negative clauses in $R^{*}_{\mathfrak{m}}(\mathcal{C})$ we get $\{\neg R(a,a)\}, \{\neg Q(a)\}$. Thus we know that $R(a,a)$, $Q(a)$ must be set to false. A slight modification of $\mathfrak{m}$ then gives a model $\mathfrak{m}_0$:

$\mathfrak{m}_0 = \{\neg R(a,a), \neg Q(a)\} \cup \{Q(s) \ / \ s \in H - \{a\}\} \cup \{R(s,t) \ / \ s \neq a \text{ or } t \neq a\} \cup$
$\{\neg P(s) \ / \ s \in H\}.$

Apparently, we did not use condensing and restricted factoring in deciding PVD. Instead we know from theorem 3.2.1. that both operations cannot be applied in a nontrivial manner on clauses in $\mathcal{C}$ (note that we did not allow condensing of true clauses). However, we will use both condensing and restricted factoring to decide the second class in this chapter. In order to obtain a decision method for both classes (simultaneously), it is reasonable to choose RSC$\mathfrak{m}$D. In [Fer90] two classes D_2, D_3 were defined and investigated which are structurally similar to KI, KII and also show a similar behaviour w.r.t. decision procedures.

Class D_2:

A set of clauses $\mathcal{C}$ belongs to D_2 if

D_2- a) $\mathcal{C}$ is Horn.

D_2-b) for all $C \in \mathcal{C}$ and for all $x \in V(C)$ it holds:

b1) $OCC(x, C_+) \leq 1$

b2) $\tau_{max}(x, C_+) \leq \tau_{min}(x, C_-)$.

Like KI, D_2 can be decided by positive hyperresolution; note that KI, D_2 are not comparable (none is contained in the other one) because in KI the restriction on the facts is stronger, for D_2 the restriction b1) also applies to rules and b2) is more restricted. D_2 does not contain DATALOG, while KI does.

Class D_3:

A set of clauses $\mathcal{C}$ belongs to D_3 if

D_3 - a) $\mathcal{C}$ is Horn.

D_3 -b) For all $C \in \mathcal{C}$ and for all $x \in V(C)$ it holds:

b1) $occ(x, C_-) \leq 1$

b2) $\tau_{max}(x, C_-) \leq \tau_{min}(x, C_+)$.

Like in the case of KI, KII, D_3 is the mirror image of D_2. But for D_3 semantic clash resolution with positive setting (= negative hyperresolution) alone does not terminate. Condensing was necessary in order to achieve this effect. Like for KI, KII, PVD we will show now that the Horn structure is not essential for the decidability of D_2, D_3.

<u>DEFINTION</u> 3.2.2.: A set of clauses $\mathcal{C}$ belongs to OCC1N (occurrence in negative part only once) if there exists a setting for $\mathcal{C}$ with the following properties:

OCC1N - 1) $OCC(x, C_{neg}) \leq 1$ for all $C \in \mathcal{C}$, $x \in V(C_{neg})$

OCC1N - 2) $\tau_{max}(x, C_{neg}) \leq \tau_{min}(x, C_{pos})$ for all $C \in \mathcal{C}$ and $x \in V(C_{neg}) \cap V(C_{pos})$.

Note that like in the case PVD, OCC1N generalizes D_2, D_3 also on Horn forms. The proof that RSCmD decides OCC1N, is substantially more technical than this for PVD. Before we start the proof of the main result we first need some technical lemmas concerning the behaviour of m.g.u.'s in resolution and factoring.

DEFINITION 3.2.3.: Let P be an atom considered as a string of symbols. Then symb(i, P) denotes the i- th symbol in P; string(i, P) denotes the string of symbols in P beginning with the i-th symbol.

DEFINITION 3.2.4.: A pair of atoms (P, Q) fulfils (∗) if there is a number k s.t. it holds:

(∗1) $\mathrm{symb}(i, P) = \mathrm{symb}(i, Q)$ for all $1 \leq i < k$.

(∗2) $OCC(x, \mathrm{string}(k, Q)) = 1$ for all $x \in V(\mathrm{string}(k, Q))$ and $V(P) \cap V(\mathrm{string}(k, Q)) = \emptyset$.

The property (∗) occurs in a natural way if we unify {P, Q} where P is an arbitrary atom and $OCC(x, Q) = 1$ for all $x \in V(Q)$.

As an example, let $R = P(x, f(x), y)$, $Q = P(f(x_1), x_2, x_3)$.
Setting $k = 3$ we get P(, P(as strings of the first two symbols from R and Q respectively and thus (∗1) , and $OCC(x_i, f(x_1), x_2, x_3) = 1$ for $x_i \in \{x_1, x_2, x_3\}$ and $V(f(x_1), x_2, x_3) \cap V(R) = \emptyset$. Computing the first mesh substituent $\{f(x_1)/x\}$ of the m.g.u. for {P, Q} we get $R' = R(f(x_1), f(f(x_1)), y)$ $Q' = Q = P(f(x_1), x_2, x_3)$. Now (∗) holds for (P', Q') with $k = 8$. Note that the number k depends on the syntax representation of atoms, but not the property (∗) itself. The length (= number of symbols) of an atom does not coincide with the corresponding notion usual in complexity theory, where only a finite alphabet is allowed. Here we work over the infinite alphabet of predicate logic.

We are now in the position to prove a structural property of most general unifiers unifying {P, Q} where (P, Q) fulfils (∗):

LEMMA 3.2.1.: Suppose that (P, Q) fulfils (∗) and that {P, Q} is unifiable. Then the m.g.u. of {P, Q} is of the form $\lambda \cup \mu$, where $\mathrm{dom}(\lambda) \subseteq V(P)$, $\lambda = \{t_1/x_1, \dots, t_k/x_k\}$ where all t_i are subterms of Q and $\mathrm{dom}(\mu) \subseteq V(Q) - V(P)$.

Remark:

Note that, in general, m.g.u.'s are not of the form expressed in lemma 3.2.1. Just take $P = P(x, f(x), f(y))$ and $Q = P(f(u), v, u)$; then the m.g.u. is

$$\sigma = \{ f^{(2)}(y)/x,\ f^{(3)}(y)/v,\ f(y)/u \}.$$

Obviously, σ cannot be represented as $\lambda \cup \mu$ in the sense of lemma 3.2.1.

Proof of lemma 3.2.1.:

By induction on $n = |V(\{P, Q\})|$.

$n = 0$: $\{P, Q\}$ is only unifiable if $P = Q$ and therefore the m.g.u. is $\emptyset$.
We only have to set $\lambda = \mu = \emptyset$.

(IH) Suppose that for all (P, Q) fulfilling $(*)$ and $|V(\{P, Q\})| = n$ the m.g.u. is of the form $\lambda \cup \mu$ as indicated above.

Case n + 1:

If $P = Q$ then $\lambda = \mu = \emptyset$ and the assertion holds.
Otherwise the disagreement set $D(P, Q)$ is $\{x, t\}$ for a variable x and a term t.
Note that we unify from left two right and (x, t) is the first pair of different corresponding terms for (P, Q).
By Robinson's unification algorithm we get
$\text{m.g.u.}(P, Q) = \{t/x\}\ \text{m.g.u.}(P\{t/x\}, Q\{t/x\})$.
But because $x \notin V(t)$ we get $|V(\{P\{t/x\}, Q\{t/x\}\})| = n$.
In order to apply (IH) we have to show first that $(P\{t/x\}, Q\{t/x\})$ fulfils $(*)$.

Case a) $x \in V(P)$.

If $x = \text{symb}(k, P)$ then by definition of the disagreement set $\text{symb}(i, P) = \text{symb}(i, Q)$ for $i < k$ and k is the maximal number with this property.
Because (P, Q) fulfils $(*)$ we get that $OCC(x, \text{string}(Q, k)) = 1$ for all $x \in V(\text{string}(Q, k))$.
Because $x \in V(P)$ we get $Q\{t/x\} = Q$ and $(P\{t/x\}, Q\{t/x\})$ fulfils $(*)$ for the same number k.

case b) $x \in V(Q)$.

Let m be the maximal number s.t. $\text{symb}(i, P\{t/x\}) = \text{symb}(i, P) = \text{symb}(i, Q\{t/x\})$. Obviously $m > k$ holds.
Because $OCC(x, \text{string}(Q,k)) = 1$ and $x = \text{symb}(k, Q)$ we get $OCC(y, \text{string}(Q\{t/x\}, m+1)) = 1$ for all y in $V(\text{string}(Q\{t/x\}, m+1))$.
Thus $(*)$ holds for $(P\{t/x\}, Q\{t/x\})$ by setting k to $m+1$.

Thus we know that $(P\{t/x\}, Q\{t/x\})$ fulfils $(*)$ in any case and we turn to $\sigma = \text{m.g.u.}(\{P\{t/x\}, Q\{t/x\}\})$.

By (IH) $\sigma = \{t_1/x_1\} \cup \dots \cup \{t_k/x_k\} \cup \mu$ where $\{x_1, \dots, x_k\} \subseteq V(P\{t/x\})$, $\text{dom}(\mu) \subseteq V(Q\{t/x\}) - V(P\{t/x\})$ and $t_1, \dots, t_k$ are subterms of Q.

1) If $x \in V(P)$ then t is a term in Q and we get
$\text{m.g.u.}(P, Q) = \{t/x\} (\{t_1/x_1\} \cup \dots \cup \{t_k/x_k\} \cup \mu)$.
Because (P, Q) fulfils $(*)$ by position k (see case a) above),
we know that $V(P) \cap V(\text{string}(k, Q)) = \emptyset$. But t is a prefix of string(k, Q) and thus $V(P) \cap V(t) = \emptyset$. Particularly $x_i \notin V(t)$ for $i = 1, \dots, k$.
By $\text{dom}(\mu) \subseteq V(Q\{t/x\}) - V(P\{t/x\})$ and $V(t) \subseteq V(P\{t/x\})$ we also have $\text{dom}(\mu) \cap V(t) = \emptyset$.

Consequently we get $\text{m.g.u.}(P, Q) = \{t/x\} \cup \{t_1/x_1\} \cup \dots \cup \{t_k/x_k\} \cup \mu$.
Moreover we have $\{x, x_1, \dots, x_k\} \subseteq V(P)$.
It remains to show $\text{dom}(\mu) \subseteq V(Q) - V(P)$. Because $\{x\} = V(P) - V(P\{t/x\})$ it is enough to show that $x \notin \text{dom}(\mu)$; the latter property is obvious, because $x \notin V(\{P\{t/x\}, Q\{t/x\}\})$.
This settles the case n+1 for $x \in V(P)$.

2) $x \in V(Q)$ and t is a term in P.
Then $\text{m.g.u}\{P,Q\} = \{t/x\}(\{t_1/x_1\} \cup \dots \cup \{t_k/x_k\} \cup \mu) =$
$= \{t_1/x_1\} \cup \dots \cup \{t_k/x_k\} \cup \{t'/x\}\mu$ for some term t'.
Set $\lambda = \{t_1/x_1\} \cup \dots \cup \{t_k/x_k\}$. We have to show that $\{x_1, .. x_k\} \subseteq V(P)$ holds; but $V(P) = V(P\{t/x\})$ and $\{x_1, \dots x_k\} \subseteq V(P\{t/x\})$ by (IH).
Thus $\text{dom}(\lambda) \subseteq V(P)$, $\lambda = \{t_1/x_1\} \cup \dots \cup \{t_k/x_k\}$ for subterms t_i in Q.

It remains to show that $\text{dom}(\mu') \subseteq V(Q) - V(P)$ for $\mu' = \{t'/x\}\mu$.
But $\text{dom}(\mu) \subseteq V(Q\{t/x\}) - V(\{P\{t/x\}) =$
$V(Q\{t/x\}) - V(P) \subseteq V(Q) - V(P)$.

By $\text{dom}(\mu') \subseteq \{x\} \cup \text{dom}(\mu)$ and by $x \in V(Q)$ we conclude $\text{dom}(\mu') \subseteq V(Q) - V(P)$. This concludes the case $n + 1$ for $x \in V(Q)$.

Q.E.D.

The following lemma shows that factoring of negative clauses in OCC1N preserves the property OCC1N and does not increase term depth.

LEMMA 3.2.2. Let $C \in \mathcal{C}$ for $\mathcal{C} \in$ OCC1N w.r.t. to a setting $\mathfrak{M}$ s.t. C is false in $\mathfrak{M}$ and let $C\vartheta$ be a factor of C. Then $\{C\vartheta\} \in$ OCC1N and $\tau(C\vartheta) \leq \tau(C)$.

Proof:

By definition of OCC1N we have $OCC(x,C) = 1$ for all $x \in V(C)$.

Furthermore we note that every factor can be computed by iterated binary factoring.

Because $\mathcal{C} \in$ OCC1N we also have $V(L) \cap V(M) = \emptyset$ for $L \neq M$ and $L, M \in C$.

If a binary factoring substitution ϑ unifies $\{L, M\}$ for $L, M \in C$ we get:

$$C\vartheta = (C - \{L,M\}) \cup \{L,M\}\vartheta.$$

Because $\tau(C - \{L,M\}) \leq \tau(C)$ anyway, it suffices to show that

$\{L\vartheta\} \in$ OCC1N and

$\tau(L\vartheta) \leq \max\{\tau(L), \tau(M)\} \quad (\leq \tau(C))$.

Let P,Q be the corresponding atom formulas to L, M.

Then (P, Q) fulfils condition (*) and by lemma 3.2.1 we have

$\vartheta = \{t_1/x_1, \dots, t_n/x_n\} \cup \mu$ for $\{x_1, \dots, x_n\} \subseteq V(P)$ $db(\mu) \subseteq V(Q)$ and the t_i are subterms of Q.

Therefore we have $P\vartheta = P\{t_1/x_1, \dots, t_n/x_n\}$.

But by $\{C\} \in$ OCC1N we know that $OCC(x,t_i) = 1$ for all $x \in V(t_i)$ and $i = 1,\dots,n$; moreover we have $V(t_i) \cap V(t_j) = \emptyset$ for all $i,j \in \{1,\dots n\}$ s.t. $i \neq j$.

Thus we get $OCC(x,P\vartheta) = 1$ for $x \in V(P\vartheta)$.

Because also (Q, P) fulfils (*) we conclude $OCC(x,Q\vartheta) = 1$ for all $x \in V(Q\vartheta)$ in the same way.

Because ϑ is a m.g.u. we also get $V(C - \{L,M\}) \cap V(L\vartheta) = \emptyset$ and by

$C - \{L,M\} \in$ OCC1N, $(C - \{L,M\}) \cup \{L\vartheta\} \in$ OCC1N.

It remains to show $\tau(N\vartheta) \leq \max\{\tau(P), \tau(Q)\}$ for $N = P, Q$.

Again, let $\vartheta = \{ t_1/x_1, \ldots, t_n/x_n\} \cup \mu$ and
$\eta = \{ t_1/x_1, \ldots, t_n/x_n \}$.

We get $P\vartheta = Q\vartheta$ and also $P\eta = Q\mu$:

We first show $\tau(P\vartheta) = \tau(P\eta) \le \max\{\tau(P), \tau(Q)\}$.
Because every x_i occurs only once in P we get

$$\tau(P\{t_i/x_i\}) \le \max\{\tau(P), \tau(Q)\} \quad \text{for every } i = 1, \ldots, n;$$

$t_i/x_i \in \vartheta$ means that (x_i, t_i) is a corresponding term pair in (P, Q) and thus $\tau_{max}(x_i, P) = \tau(x_i, P) = \tau(t_i, Q)$. Because x_i is only substituted at the place of the corresponding pair we get

$$\tau(P\{t_i/x_i\}) \le \max\{\tau(P), \tau(Q)\}$$

(if $\tau(P) \le \tau(Q)$ in advance then $\tau(P\{t_i/x_i\}) \le \tau(Q)$).

But either $\tau(P\eta) = \tau(P)$ and thus $\tau(P\eta) \le \max\{\tau(P), \tau(Q)\}$ or
$\tau(P\eta) = \tau(x_i, P) + \tau(t_i)$ for some $i \in \{1, \ldots n\}$
(note that by $OCC(x_i, P) = 1$ $\tau_{max}(x_i, P) = \tau(x_i, P)$). But

$$\tau(x_i, P) + \tau(t_i) = \tau(P\{t_i/x_i\}) \le \max\{\tau(P), \tau(Q)\}$$

by the argumentation above.

It follows

$$\tau(P\vartheta) = \tau(P\eta) \le \max\{\tau(P), \tau(Q)\}.$$

But $Q\vartheta = P\vartheta$ and thus also

$$\tau(Q\vartheta) \le \max\{\tau(P), \tau(Q)\}.$$

We get $\tau(C\vartheta) = \tau((C - \{L, M\}) \cup \{P\vartheta\})$ and by
$\tau(\{L, M\}\vartheta) = \tau(L\vartheta) = \tau(P\vartheta) \le \max\{\tau(P), \tau(Q)\}$
$\tau(C\vartheta) = \tau(C)$.

Q.E.D.

In the next lemma we show that OCC1N is closed under binary semantic resolution and that the term depth of the resolvents cannot increase.

<u>LEMMA</u> 3.2.3.: Let $\{C, D\} \in$ OCC1N w.r.t. a setting $\mathfrak{m}$; let C be positive in $\mathfrak{m}$, D be negative in $\mathfrak{m}$ and R be a binary $\mathfrak{m}$-resolvent of C and D. Then $\{R\} \in$ OCC1N and $\tau(R_{neg}) \leq \max\{\tau(C_{neg}), \tau(D)\}$.

<u>Proof:</u>

Let $C = C_{neg} \cup C_{pos}$, $L \in C_{pos}$ and $M \in D$ s.t. ϑ is a m.g.u. of $\{L, M^d\}$ and $R = C_{neg}\vartheta \cup (D - \{M\})\vartheta \cup (C_{pos} - \{L\})\vartheta$ is a binary semantic resolvent of C and D.

Because $R_{neg} = C_{neg}\vartheta \cup (D - \{M\})\vartheta$ we have to show

$$\tau(C_{neg}\vartheta \cap (D - \{M\})\vartheta) \leq \max\{\tau(C_{neg}), \tau(D)\}.$$

Because $\{D\} \in$ OCC1N, it is variable - decomposed; because ϑ is a m.g.u. we have $dom(\vartheta) \cap V(D - \{M\}) = \emptyset$ and thus $(D - \{M\})\vartheta = (D - \{M\})$. Because, trivially, $\tau(D - \{M\}) \leq \tau(D)$ it suffices to show $\tau(C_{neg}\vartheta) \leq \max\{\tau(C_{neg}), \tau(D)\}$. We further reduce the unequality to be proved to

$$\tau(C_{neg}\vartheta) \leq \max\{\tau(C_{neg}), \tau(M)\}.$$

Because M is a literal from a false clause, we get that (P, Q) fulfils $(\bullet)$ for $P = at(L)$, $Q = at(M)$.

By lemma 3.2.1. we know that $\vartheta = \{t_1/x_1, \dots, t_n/x_n\} \cup \mu$ where $\{x_1, \dots, x_n\} \subseteq V(P)$, t_i are subterms of Q and $dom(\mu) \subseteq V(Q)$.
It follows that $C_{neg}\vartheta = C_{neg}\sigma$ for $\sigma = \{t_1/x_1, \dots, t_n/x_n\}$.

If $\tau(C_{neg}\sigma) = \tau(C_{neg})$ then $\tau(C_{neg}\vartheta) = \tau(C_{neg}\sigma) \leq \max\{\tau(C_{neg}), \tau(M)\}$ and we have what we want.

If $\tau(C_{neg}\sigma) > \tau(C_{neg})$ then there is an $i \in \{1, \dots, n\}$ s.t.

(A) $\tau(C_{neg}\sigma) = \tau_{max}(x_i, C_{neg}) + \tau(t_i)$ for an $x_i \in V(C_{neg}) \cap V(C_{pos})$.
But t_i is a term which occurs in M and therefore

(B) $\tau(t_i) \leq \tau(M) - \tau_{min}(x_i, L)$
(note that, by $t_i/x_i \in \lambda$, (x_i, t_i) is a corresponding pair in (P, Q)).

But by condition OCC1N) - 2) and $x_i \in V(C_{neg}) \cap V(C_{pos})$ we get

$$\tau_{max}(x_i, C_{neg}) \le \tau_{min}(x_i, C_{pos})$$

and also $\tau_{min}(x_i, C_{pos}) \le \tau_{min}(x_i, L)$.

Therefore: (C) $\tau_{max}(x_i, C_{neg}) \le \tau_{min}(x_i, L)$.

Combining (A) and (C) we get the inequality

$$\tau(C_{neg}\sigma) \le \tau_{min}(x_i, L) + \tau(t_i)$$

and using (B):

$$\tau(C_{neg}\sigma) \le \tau_{min}(x_i, L) + \tau(M) - \tau_{min}(x_i, L) = \tau(M).$$

Thus, in any case $\tau(C_{neg}\sigma) \le \max\{\tau(C_{neg}), \tau(M)\}$ and we have proved

$$\tau(R_{neg}) \le \max\{\tau(C_{neg}), \tau(D)\}.$$

It remains to show that $\{R_{neg}\} \in$ OCC1N holds.

For this purpose we have to show first that $OCC(x, R_{neg}) = 1$ for all $x \in R_{neg}$.

But $R_{neg} = C_{neg}\vartheta \cup (D - \{M\})$, where $D - \{M\} \in$ OCC1N and
$V(C_{neg}\vartheta) \cap V(D - \{M\}) = \emptyset$.

Therefore it is sufficient to prove $OCC(x, C_{neg}\vartheta) = 1$ for all $x \in V(C_{neg}\vartheta)$.

But $C_{neg}\vartheta = C_{neg}\sigma$ for $\sigma = \{t_1/x_1, \dots, t_n/x_n\}$ where $OCC(x, t_i) = 1$ for all $x \in V(t_i)$, $i \in \{1, \dots, n\}$ and $V(t_i) \cap V(t_j) = \emptyset$ for $i, j \in \{1, \dots, n\}$ and $i \neq j$.

It follows immediately that, by $C \in$ OCC1N, $OCC(x, C_{neg}\sigma) = 1$ for all $x \in V(C_{neg}\sigma)$.

We get that OCC1N - 1) holds for R_{neg}.

To show OCC1N - 2) we have to prove that

$$\tau_{max}(x, R_{neg}) \le \tau_{min}(x, R_{pos}) \text{ for all } x \in V(R_{neg}) \cap V(R_{pos}).$$

But $R_{neg} = C_{neg}\vartheta \cup (D - \{M\})$ where $V(D - \{M\}) \cap V(R_{pos}) =$
$= V(D - \{M\}) \cap V((C_{pos} - \{L\})\vartheta) = \emptyset$

(note that $V(C) \cap V(D) = \emptyset$ before resolution and $dom(\vartheta) \cap V(D - \{M\}) = \emptyset$).

Thus it suffices to prove

$$\tau_{max}(x, C_{neg}\vartheta) \le \tau_{min}(x, (C_{pos} - \{L\})\vartheta) \quad \text{for all}$$
$$x \in V(C_{neg}\vartheta) \cap V((C_{pos} - \{L\})\vartheta).$$

Suppose now that $x \in V(C_{neg}\vartheta) \cap V((C_{pos} - \{L\})\vartheta)$.
If $x \in V(C_{neg}) \cap V((C_{pos} - \{L\}))$ and $x \notin dom(\vartheta)$ then
$x \in V(C_{neg}\vartheta) \cap V((C_{pos} - \{L\})\vartheta)$ and

a) $\tau_{max}(x, C_{neg}\vartheta) = \tau_{max}(x, C_{neg})$,
b) $\tau_{min}(x, (C_{pos} - \{L\})\vartheta) = \tau_{min}(x, C_{pos} - \{L\})$.

Note that $C\vartheta = C\sigma$ for $\sigma = \{t_1/x_1, \dots, t_n/x_n\}$, where the t_i are terms in M. Because $V(t_i) \cap V(C) = \emptyset$ for all i, the variable x cannot occur in $rg(\sigma)$. We thus have a guarantee that x is not introduced additionally via ϑ and the occurrences of x in $C_{neg}\vartheta$ are those of C_{neg}; this yields the properties (a) and (b).

By OCC1N - 2 for C we conclude

$$\tau_{max}(x, C_{neg}\vartheta) = \tau_{max}(x, C_{neg}) \le \tau_{min}(x, C_{pos}) \le \tau_{min}(x, C_{pos} - \{L\}) =$$
$$= \tau_{min}(x, (C_{pos} - \{L\})\vartheta).$$

If $x \in V(C_{neg}) \cap V(C_{pos} - \{L\})$ and $x \in dom(\vartheta)$ then, as m.g.u.'s are idempotent, $x \notin V(C_{neg}\vartheta) \cap V((C_{pos} - \{L\})\vartheta)$.
If $x \notin V(C_{neg}) \cap V(C_{pos} - \{L\})$ then x must occur in $rg(\vartheta)$.
Because $C_{neg}\vartheta = C_{neg}\sigma$ for $\sigma = \{t_1/x_1, \dots, t_n/x_n\}$ and $dom(\sigma) \subseteq V(L)$, x must occur in some of the t_i. Because the t_i are mutually variable - disjoint, x can only occur in one of the t_i's.

Thus suppose $x \in V(t_i)$.
Because $x \notin V(C_{neg}) \cap V(C_{pos} - \{L\})$ we get

(1) $\tau_{max}(x, C_{neg}\vartheta) = \tau_{max}(x_i, C_{neg}) + \tau_{max}(x, t_i)$ and
(2) $\tau_{min}(x, (C_{pos} - \{L\})\vartheta) = \tau_{min}(x_i, C_{pos} - \{L\}) + \tau_{min}(x, t_i)$.

But as t_i is a term occuring in M, $OCC(x, t_i) = 1$ and

(3) $\tau_{max}(x, t_i) = \tau_{min}(x, t_i)$

By OCC1N - 2) we know that

$$\tau_{max}(x_i, C_{neg}) \le \tau_{min}(x_i, C_{pos})$$

and therefore

(4) $\tau_{max}(x_i, C_{neg}) \le \tau_{min}(x_i, C_{pos} - \{L\})$

combining (1) - (4) we easily get

$$\tau_{max}(x, C_{neg}\vartheta) \le \tau_{min}(x, (C_{pos} - \{L\})\vartheta).$$

Q.E.D.

We are now in the position to show that OCC1N is closed under SC$\mathfrak{m}$ - resolution and that the term depth of clash resolvents cannot increase.

<u>LEMMA</u> 3.2.4.: Let $\mathcal{C}$ be a set of clauses in OCC1N w.r.t. the setting $\mathfrak{m}$ and $\Gamma = (C;\ D_1, \dots, D_n)$ be a semantic clash of clauses in OCC1N. If R is a clash resolvent of Γ then $\{R\} \in$ OCC1N and $\tau(R) \leq \max\{\tau(C), \tau(D_1), \dots, \tau(D_n)\}$.

<u>Proof</u>:

If R is a clash resolvent of Γ then R is R_n for some R_n defined as:

$R_0 = C$

R_{i+1} = a binary semantic resolvent of R_i and (a factor of) D_{i+1} for $i < n$.

We prove by induction on i that $\tau((R_i)_{neg}) \leq \max\{\tau(C), \tau(D_1), \dots, \tau(D_n)\}$ and $\{R_i\} \in$ OCC1N.

$i = 0$: $R_0 = C$ and $\{C\} \in$ OCC1N; moreover, $\tau(C_{neg}) \leq \tau(C)$.

Suppose that the assertion holds for i.

If $i = n$ we are done.

Thus assume that $i < n$.

Then R_{i+1} is a binary resolvent of R_i and D_{i+1}', where $D_{i+1}' = D_{i+1}$ or $D_{i+1}' = D_{i+1}\eta$ for a factoring substitution η.

Because $\{D_{i+1}\} \in$ OCC1N , we know by lemma 3.2.2. that $D_{i+1}' \in$ OCC1N and $\tau(D_{i+1}') = \tau(D_{i+1})$.

By lemma 3.2.3. we get

$\tau((R_{i+1})_{neg}) \leq \max\{\tau((R_i)_{neg}), \tau(D_{i+1}')\} \leq \max\{\tau(R_i)_{neg}, \tau(D_{i+1})\}$ and $\{R_{i+1}\} \in$ OCC1N.

Applying the induction hypothesis we thus obtain

$\tau((R_{i+1})_{neg}) \leq \max\{\tau(C), \tau(D_1), \dots, \tau(D_n)\}$.

But because R_n is a clash resolvent we also have $(R_n)_{neg} = R_n$ and therefore

$\tau(R) = \tau(R_n) \leq \max\{\tau(C), \tau(D_1), \dots, \tau(D_n)\}$.

Q.E.D.

<u>THEOREM</u> 3.2.2.: SC$\mathfrak{m}$D - resolution is a decision procedure for OCC1N. More precisely:
If $\mathcal{C} \in$ OCC1N then there is a setting $\mathfrak{m}$ s.t. $R^{*}_{\mathfrak{m}}(\mathcal{C})$ is finite (we take this $\mathfrak{m}$ which fulfils OCC1N)-1) OCC1N)-2)). Moreover there is an algorithm defining for every $\mathcal{C} \in$ OCC1N a setting $\mathfrak{m}$ s.t. $R^{*}_{\mathfrak{m}}(\mathcal{C})$ is finite.

<u>Proof</u>:
We first show that $R^{*}_{\mathfrak{m}}(\mathcal{C}) \in$ OCC1N for $\mathfrak{m}$ s.t. $\mathcal{C} \in$ OCC1N w.r.t. $\mathfrak{m}$.
Because $\bigcup_{i=0}^{\infty} R^{i}_{\mathfrak{m}}(\mathcal{C}) = R^{*}_{\mathfrak{m}}(\mathcal{C})$ we prove $R^{i}_{\mathfrak{m}}(\mathcal{C}) \in$ OCC1N by induction on i.

$R^{0}_{\mathfrak{m}}(\mathcal{C}) = \mathcal{C}_0 = \{C \mid C \in \mathcal{C}, C \text{ true in } \mathfrak{m}\} \cup \{C_{cond} \mid C \in \mathcal{C}, \mathcal{C} \text{ false in } \mathfrak{m}\}$.
Because for every false C, OCC1N)-1) holds, also for the condensation (because it is valid for any factoring) we get $\mathcal{C}_0 \in$ OCC1N.
(IH) Suppose that $R^{i}_{\mathfrak{m}}(\mathcal{C}) \in$ OCC1N.

Then $R^{i+1}_{\mathfrak{m}}(\mathcal{C}) = (R^{i}_{\mathfrak{m}}(\mathcal{C}) \cup \text{RSC}\mathfrak{m}\text{D}(R^{i}_{\mathfrak{m}}(\mathcal{C})))/\sim_v$.

By (IH) and lemma 3.2.4. we know that $\text{RSC}\mathfrak{m}(R^{i}_{\mathfrak{m}}(\mathcal{C})) \in$ OCC1N.
But every clause $C \in \text{RSC}\mathfrak{m}(R^{i}_{\mathfrak{m}}(\mathcal{C}))$ is false and in OCC1N.

By lemma 3.2.2., every factor of C is in OCC1N; we conclude that also $C_{cond} \in$ OCC1N; thus $\text{RSC}\mathfrak{m}\text{D}(R^{i}_{\mathfrak{m}}(\mathcal{C})) \in$ OCC1N.
So we get $R^{i+1}_{\mathfrak{m}}(\mathcal{C}) \in$ OCC1N and, by induction, $R^{*}_{\mathfrak{m}}(\mathcal{C}) \in$ OCC1N.

It remains to show, that $R^{*}_{\mathfrak{m}}(\mathcal{C})$ is finite:
By an induction argument analogous to that before we conclude $\tau(R^{i}_{\mathfrak{m}}(\mathcal{C})) \leq \tau(\mathcal{C})$ for all i (we just use lemma 3.2.4 and the fact that condensed clauses cannot be of higher depth than uncondensed ones).
But $R^{*}_{\mathfrak{m}}(\mathcal{C}) - \mathcal{C}$ consists of negative clauses only. By definition of OCC1N, every $C \in R^{*}_{\mathfrak{m}}$ is decomposed (even after condensation). But there can be only finitely many decomposed, condensed clauses of finite fixed depth (note that there may be no literals which are variants of other literals in C).
It follows that $R^{*}_{\mathfrak{m}}(\mathcal{C})$ is finite.

Finally we observe that $\mathfrak{m}$ can be found effectively, as in the case PVD: There are only finitely many settings possible for $\mathcal{C}$ and OCC1N) -1), OCC1N) - 2) is decidable for every $\mathfrak{m}$.

Q.E.D.

SC$\mathfrak{m}$D - resolution can be used as decision method for both PVD and OCC1N (thus for PVD $\cup$ OCC1N). If $\mathcal{C} \in$ PVD $\cup$ OCC1N we find some setting $\mathfrak{m}$ s.t. $\mathcal{C} \in$ PVD w.r.t. $\mathfrak{m}$ or $\mathcal{C} \in$ OCC1N w.r.t. $\mathfrak{m}$ (possibly in both classes) and may compute $R^{*}_{\mathfrak{m}}(\mathcal{C})$.
Rather than just a decision "procedure" it is a decision method by which the resolution procedure corresponding to a setting $\mathfrak{m}$ can be found algorithmically. This approach is somewhat different from Joyner's, where a single refinement is specified to decide some classes. But similarly we only define complete resolution refinements, which at the same time can be used as ordinary theorem provers. Even if $\mathcal{C} \notin$ PVD $\cup$ OCC1N we may try to find a setting s.t. as many clauses as possible fulfil the conditions PVD1), PVD2) or OCC1N - 1), OCC1N) - 2). Although we can no longer guarantee termination, the procedure favours the production of clauses having low term depth. In this sense it can provide a method to control growth of nesting based on the logical structure of the set of input clauses rather than a method based on heuristics only.

In case of the Bernays - Schönfinkel class (BS), the class PVD can be used to obtain an efficient decision algorithm.

Let BS = $\{(\exists \bar{x})(\forall \bar{y}) M(\bar{x}, \bar{y}) \mid M(\bar{x}, \bar{y})$ is a function - free matrix with variable vectors $\bar{x}, \bar{y}\}$.

Then the clause form of BS is BS* = $\{\mathcal{C} \mid \tau(\mathcal{C}) = 0\}$.
BS* is easily decidable by a ground method because for every $\mathcal{C} \in$ BS* the Herbrand universe is finite.
A straight forward decision procedure is the following:

1) Compute all ground instances of clauses in $\mathcal{C}$.

2) Apply the Davis - Putnam procedure [DP 60].

This method, although easily defined, can be very expensive because the number of ground instances can be very high (for every clause C we obtain $|H(\mathcal{C})|^{|V(c)|}$ ground instances).
Thus resolution on $\mathcal{C}$ itself might be much more promising. But unfortunately BS* cannot be decided by semantic clash resolution at all.

<u>EXAMPLE</u> 3.2.3.:

$\mathcal{C} = \{\{P(x,z,u),\ \neg P(x,y,u),\ \neg P(y,z,u)\},$

$\{P(x,x,a)\},$

$\{\neg P(x,z,u), P(x,y,u),\ P(y,z,u)\}$

$\{\neg P(x,x,b)\}\}$ (a, b are different constant symbols).

As setting we only have negative setting ($\mathfrak{m}_n$) and positive setting ($\mathfrak{m}_p$). Under RSC$\mathfrak{m}_p$D we get an infinite sequence of clauses generated by the first and the fourth clause, which cannot be eliminated by condensing or by subsumption. For RSC$\mathfrak{m}_n$D a similar thing happens with the second and the third clause. This proves that no semantic refinement can decide BS*. Obviously neither $\mathfrak{m}_p$ nor $\mathfrak{m}_n$ fulfils conditions PVD1), PVD2) and therefore $\mathcal{C} \notin$ PVD.
However we will define a simple method to transform a set of clauses $\mathcal{C}$ in BS* into a set of clauses $\mathcal{C}' \in$ PVD s.t. $\mathcal{C}$ and $\mathcal{C}'$ are sat-equivalent. We proceed as follows:

case a) $\mathcal{C} \in$ PVD: $\mathcal{C}' = \mathcal{C}$.

case b) $\mathcal{C} \notin$ PVD:

b1) choose a setting $\mathfrak{m}$ for $\mathcal{C}$.

b2) For all $C \in \mathcal{C}$ we define:
If $V(C_{neg}) \subseteq V(C_{pos})$ then $T(C) = \{C\}$.
If $V(C_{neg}) - V(C_{pos}) \neq \emptyset$ then
$T(C) = \{C\lambda \mid dom(\lambda) = V(C_{neg}) - V(C_{pos}), rg(\lambda) \subseteq H(\mathcal{C})\}$.

b3) $\mathcal{C}' = \bigcup_{C \in \mathcal{C}} T(C)$

In both cases a), b) $\mathcal{C}' \in$ PVD and $\mathcal{C}'$ is sat-equivalent to $\mathcal{C}$.
After definition of $\mathcal{C}'$ we only need to compute $R^{*}_{\mathfrak{m}}(\mathcal{C}')$.
The algorithm above can be optimized by searching for an $\mathfrak{m}$ s.t. $\mathcal{C}'$ becomes minimal (this can be done without actually computing $\mathcal{C}'$). In general the size of $\mathcal{C}'$ will be much smaller than the set of all ground instances of $\mathcal{C}$; in such a case the method above is clearly superior to the ground method. We return to example 3.2.3.:

$\mathcal{C} \notin$ PVD, so we choose $\mathfrak{m}_p$ and compute $\mathcal{C}'$.

$\mathcal{C}' = \{\{P(x,z,u),\ \neg P(x,a,u),\ \neg P(a,z,u)\},$

$\{ P(x,z,u),\ \neg P(x,b,u),\ \neg P(b,z,u) \}$,

$\{ P(x,x,a) \}$,

$\{ \neg P(x,z,u),\ P(x,y,u),\ P(y,z,u) \}$,

$\{ \neg P(a,a,b)\},\ \{ \neg P(b,b,b) \} \}$.

$|\mathcal{C}'| = |\mathcal{C}| + 2$ and $\mathcal{C}' \in$ PVD.

For practical purposes we may apply subsumption; because $SC\mathfrak{m}$-resolution + subsumption is complete and $RSC\mathfrak{m}$ decides PVD, so $RSC\mathfrak{m}$ + subsumption also does. Let $sub(\mathcal{C})$ the subset of $\mathcal{C}$ obtained by $\mathcal{C}$ after deleting subsumed clauses; instead of $R^{*}_{\mathfrak{m}}$ we define $R^{*}_{\mathfrak{m},s}$ as follows:

$$R^{0}_{\mathfrak{m},s}(\mathcal{C}) = sub(\mathcal{C}),$$
$$R^{i+1}_{\mathfrak{m},s}(\mathcal{C}) = sub(R^{i}_{\mathfrak{m},s}(\mathcal{C}) \cup RSC\mathfrak{m}(R^{i}_{\mathfrak{m},s}(\mathcal{C}))),$$
$$R^{*}_{\mathfrak{m},s}(\mathcal{C}) = \bigcup_{i \in N} R^{i}_{\mathfrak{m},s}(\mathcal{C}).$$

By completeness under subsumption $\square \in R^{*}_{\mathfrak{m},s}(\mathcal{C})$ iff $\square \in R^{*}_{\mathfrak{m}}(\mathcal{C})$.

We now compute $R^{*}_{\mathfrak{m}_p,s}(\mathcal{C}')$

$sub(\mathcal{C}') = \mathcal{C}'$

$RSC\mathfrak{m}_p(\mathcal{C}') = \{\{ \neg P(a,a,b)\},\ \{\neg P(b,b,b)\},\ \{ \neg P(a,b,b),\ \neg P(b,a,b)\}\}$ and

$\mathcal{C}_1' = sub(\mathcal{C}' \cup RSC\mathfrak{m}_p(\mathcal{C}')) = \mathcal{C}' \cup \{\{ \neg P(a,b,b),\ \neg P(b,a,b)\}\}$.

$RSC\mathfrak{m}_p(\mathcal{C}_1') = RSC\mathfrak{m}_p(\mathcal{C}') \cup \{\{\neg P(b,a,b),\ \neg P(a,b,b),\ \neg P(b,b,b)\},$

$\{\neg P(a,b,b),\ \neg P(b,a,b),\ \neg P(a,a,b)\}\}$.

$\mathcal{C}_2' = sub(\mathcal{C}_1' \cup RSC\mathfrak{m}_p(\mathcal{C}_1')) = \mathcal{C}_1'$.

Therefore $R^{*}_{\mathfrak{m}_p,s}(\mathcal{C}') = \mathcal{C}_1'$ and $\square \notin \mathcal{C}_1'$; we conclude that $\mathcal{C}'$ (and thus $\mathcal{C}$) is satisfiable.

Using the ground saturation method for the Bernays-Schönfinkel class we obtain a set of 36 ground clauses which have to be tested for satisfiability.

3.3 UNDECIDABLE PROBLEMS

It is natural question, whether the classes PVD and OCC1N are in some sense "arbitrary". First of all PVD contains DATALOG and, like OCC1N, gives a simple clause syntax criterion for decidability. Moreover the classes are quite sharp w.r.t. undecidability. We show this in the case of PVD.

As defined in chapter 3.2. the class KII, defined as: $\mathcal{C} \in$ KII iff

a) $\mathcal{C}$ is Horn,

b) $V(C_-) \subseteq V(C_+)$ and $\tau(C_-) = 0$ for rules $C \in \mathcal{C}$,

c) Goals are ground,

is a subclass of PVD (restriction to positive setting and Horn).

We show now that by introducing the clause $T = \{P(x,z), \neg P(x,y), \neg P(y,z)\}$ (transitivity) into KII we get an undecidable class; while $\tau(T_-) = 0$ we do not have $V(T_-) \subseteq V(T_+)$ (violated by the variable y).

Let KII' = $\{\mathcal{C} \cup \{T\} \,/\, \mathcal{C} \in \text{KII}\}$ ([Lei90]). We show that every equational theory (axiomatized in predicate logic) belongs to KII':

REF) $\{P(x,x)\}$.

SYMM) $\{\neg P(x,y), P(y,x)\}$.

T) $\{\neg P(x,y), \neg P(y,z), P(x,z)\}$.

SUBfi) $\{\neg P(x,y), P(f(x_1, \dots, x_{i-1}, x, x_{i+1}, \dots, x_n), f(x_1, \dots, x_{i-1}, y, x_{i+1}, \dots x_n))\}$
for every function symbol f and every position i in f.

EQ1) $\{P(s_1, t_1)\}$

EQn) $\{P(s_n, t_n)\}$

(EQ1) - EQn) are arbitrary equations; s_i, t_j are terms.

¬CON) $\{\neg P(w_1, w_2)\}$, where w_1, w_2 are ground terms.

All axioms and ¬CON) are in KII', and with exception of T also in KII. The clauses encode the word problem of the equational theory $\{s_1 = t_1, \dots, s_n = t_n\}$, $\neg P(w_1, w_2)$ codes the negation of the equation $v_1 = v_2$ where w_1, w_2 are defined from v_1, v_2 by substituting every variable by a (new) Skolem constant.
It is well known that there are equational theories with undecidable word problems (just take the theory of combinators with η - equality [ST71]).
Thus we see that KII' is undecidable.

It was shown in [Fer91], that the restriction $V(C-) \subseteq V(C+)$ is not only important for Horn - but also for Horn + Krom classes.

Let KIIK be the following class of set of clauses $\mathcal{C}$:
(a) $\mathcal{C}$ is Horn and Krom.
(b) $\tau(C_-) = 0$ for every $C \in \mathcal{C}$.
(c) Goals in $\mathcal{C}$ are ground.

It is obvious, that by adding the condition $V(C-) \subseteq V(C+)$ we get a proper subclass of KII.
We show now that KIIK is an undecidable class.

It was proved in [SS 88], that the class KH2, consisting of all sets of clauses of the form $\mathcal{C} = \{C_1, C_2, F_1, G_1\}$, where C_1, C_2 are two Krom rules, F_1 is a ground fact and G_1 is a ground goal, is undecidable.
To show our result, we construct for every $\mathcal{C} \in$ KH2 a set $\mathcal{C}' \in$ KIIK (by an effective construction) s.t. $\mathcal{C}$ and $\mathcal{C}'$ are sat-equivalent.
The construction works as follows:
Let $C \in \{C_1, C_2\}$ for $\mathcal{C} = \{C_1, C_2, F_1, G_1\}$ as above and let

$$C = \{P(t_1, \dots, t_n), \neg Q(s_1, \dots, s_m)\}.$$

We define

$$T(C) = \Big\{\{P(t_1, \dots, t_n), \neg H_C(x_1, x_1, x_2, x_2, \dots, x_m, x_m, v_1, v_2, \dots, v_k)\}, \\ \{H_C(s_1, y_1, s_2, y_2, \dots, s_m, y_m, v_1', \dots, v_k'), \neg Q(y_1, \dots, y_m)\}\Big\} \text{ where } \\ V(C+) \cap V(C-) = \{v_1, \dots, v_k\}.$$

H_C is a new predicate symbol which only occurs in the clauses of $T(C)$.
Observe, that by resolving the two clauses in $T(C)$ we get C back. Thus from $\mathcal{C}' = T(C_1) \cup T(C_2) \cup \{F_1\} \cup \{G_1\}$ we can derive $\mathcal{C}$, and $\mathcal{C}' \in$ KIIK.
It remains to show, that the satisfiability of $\mathcal{C}$ implies that of $\mathcal{C}'$.

Suppose that $\mathcal{C}'$ is unsatisfiable. Then there is a lock refutation of $\mathcal{C}'$. By giving priority to the literals $H_C(\dots)$ and $\neg H_C(\dots)$ in indexing, we get, that all clauses not containing atoms H_{C_1}, H_{C_2} are also derivable from $\{C_1, C_2, F_1, G_1\}$.
Therefore $\square \in R^*(\mathcal{C}')$ iff $\square \in R^*(\mathcal{C})$ and $\mathcal{C}$, $\mathcal{C}'$ are sat-equivalent.
Thus we have shown that KIIK is undecidable.

In KIIK, by the T(C) construction, it is sufficient that two clauses violate $V(C_-) \subseteq V(C_+)$ in order to get undecidability. In KII', by the more general clause form, one clause (it is T) suffices not fulfilling $V(C_-) \subseteq V(C_+)$. Thus it is apparent that KII, and thus also PVD, is quite sharp w.r.t. to the structural property of variable occurrence.

3.4 HORN CLASSES

For the classes in chapter 3.2. we had to focus on two measures for guaranteeing termination, clause size and term depth.
Restricting the propositional structure of clauses to Horn, we can get rid of the clause size-problem and concentrate on term depth only. The reason is, that under negative setting the only false Horn clauses are unit clauses. Applying hyperresolution w.r.t. $\mathfrak{m}_n$ (negative setting) we only derive positive unit clauses and thus $R^*_{\mathfrak{m}_n}(\mathcal{C})$ is finite if there is a bound on term depth.
According to [CL 73] we call $SC\mathfrak{m}_n$-resolution positive hyperresolution because it produces positive clauses only (in the syntactical sense).
It is almost trivial to verify that the Bernays-Schönfinkel class, restricted to a Horn matrix can be decided by $SC\mathfrak{m}_n$-resolution. For this purpose we define:

BSH = $\{(\exists \bar{x})(\forall \bar{y}) M(\bar{x},\bar{y}) \mid M(\bar{x},\bar{y})$ is a function free matrix in Horn form with free variable vectors $\bar{x},\bar{y}\}$.

By Skolemization we get:

$BSH^* = \{\mathcal{C} \mid \mathcal{C}$ is a set of Horn clauses and $\tau(\mathcal{C}) = 0\}$.

Let $\mathcal{C}$ be in BSH^*. Then $R^*_{\mathfrak{m}_n}(\mathcal{C}) - \mathcal{C}$ consists of positive unit clauses only; because $R^*_{\mathfrak{m}_n}(\mathcal{C})$ does not contain any function symbols, all $C \in R^*_{\mathfrak{m}_n}(\mathcal{C})$ fulfil $\tau(C) = 0$. Let $C \sim_v D$ if D is a variant of C ($C\sigma = D$ for a permutation substitution σ). Clearly $R^*_{\mathfrak{m}_n}(\mathcal{C})/\sim_v$ (the class $R^*_{\mathfrak{m}_n}(\mathcal{C})$ factored under the equivalence relation $\sim_v$) must be finite. Note that we can avoid factoring w.r.t.$\sim_v$ if we keep all clauses in a variable-standard form.
Thus positive hyperresolution decides BSH^* and therefore also DATALOG which is a subclass of BSH^*.

Although we cannot deal with the Bernays-Schönfinkel class using semantic clash resolution only (we refer to the end of chapter 3.2), we can generalize BSH* to classes containing function symbols.
The following class was discussed in [Fer90] (there it was named D_1):

DEFINITION 3.4.1.: VED (variables in equal depth) is the set of all sets of Horn clauses $\mathcal{C}$ s.t. it holds:
For all $C \in \mathcal{C}$ and for all $x \in V(C)$ it holds $\tau_{min}(x, C) = \tau_{max}(x, C)$ (that means, every variable occurs only at a certain fixed depth within a clause).
In BSH* all variables occur at depth 0 and thus trivially BSH* $\subseteq$ VED. Note that in different clauses, the depth of the variables may be different; thus $\{\{\neg P(f(x)), \neg P(g(x))\}, \{\neg Q(x,y), Q(y,x)\}, \{P(a), Q(f(x))\}\} \in$ VED.
We show now, that $\mathrm{RSC}\mathfrak{m}_n$ is a decision procedure for VED.
The first step consists in showing that (binary) semantic resolution cannot increase the maximal term depth.

LEMMA 3.4.1.: Let C be a goal or a rule and D be a fact (C is positive in $\mathfrak{m}_n$ and D is negative in $\mathfrak{m}_n$) and let R be a (semantic) resolvent of C, D. Then $\tau(R) \leq \max\{\tau(C), \tau(D)\}$ and
$\{R\} \in$ VED if $\{C, D\} \in$ VED.

Proof:
w.l.o.g. we may assume that C and D are variable disjoint. Let σ be the m.g.u. of the resolvent of C and D. Then σ can be written as concatenation of mesh substituents in the form $\sigma = \{t_1/x_1\} \dots \{t_n/x_n\}$. The substitution t_i/x_i corresponds to the i-th step of the Robinson unification algorithm from left to right.

Let $\varphi_0 = \varepsilon$

$\varphi_{i+1} = \varphi_i\{t_{i+1}/x_{i+1}\}$

Then $\varphi_n = \sigma$ and, by $\tau(R) = \tau((C - M)\sigma) \leq \tau(C\sigma)$ where M is the resolved literal in C, it is sufficient to show by induction on i:

A(i)-1) $\tau(x, C\varphi_i) = \tau(x, D\varphi_i)$ for $x \in V(C\varphi_i) \cap V(D\varphi_i)$

$\tau_{max}(x, C\varphi_i) = \tau_{min}(x, C\varphi_i)$ for $x \in V(C\varphi_i) - V(D\varphi_i)$

$\tau_{max}(x, D\varphi_i) = \tau_{min}(x, D\varphi_i)$ for $x \in V(D\varphi_i) - V(C\varphi_i)$

A(i)-2) $\max\{\tau(C\varphi_i), \tau(D\varphi_i)\} \leq \max\{\tau(C), \tau(D)\}$

$A(0)$: obvious by $V(C) \cap V(D) = \emptyset$, $\{C, D\} \in$ VED, and by $\varphi_0 = \varepsilon$.

(IH) Suppose that $A(i)$ holds.
Then $\varphi_{i+1} = \varphi_i\{t_{i+1}/x_{i+1}\}$. Because $\{x_{i+1}, t_{i+1}\}$ is a disagreement set for $\{M\varphi_i, P\varphi_i\}$ (for $D = \{P\}$) t_{i+1} is either a term in $M_{\varphi i}$ or in $P\varphi_i$.
Suppose that $SUB(k, M\varphi_i) = x_{i+1}$ and $SUB(k, P\varphi_i) = t_{i+1}$.
By (IH) all variables in t_{i+1} can only occur in some fixed depth and thus also $A(i+1)-1)$ holds for $C\varphi_{i+1}$, $D\varphi_{i+1}$.
The argumentation is the same for $SUB(k, M\varphi_i) = t_{i+1}$, $SUB(k, P\varphi_i) = x_{i+1}$.

ad $A(i+1) - 2$:
Again suppose w.l.o.g. that $SUB(k, M\varphi_i) = x_{i+1}$ and $SUB(k, P\varphi_i) = t_{i+1}$.
Then either $\tau(C\varphi_{i+1}) = \tau(C\varphi_i\{t_{i+1}/x_{i+1}\}) = \tau(C\varphi_i)$ in which case we have

$$\max\{\tau(C\varphi_{i+1}), \tau(D\varphi_{i+1})\} = \max\{\tau(C\varphi_i), \tau(D\varphi_i)\} \leq \max\{\tau(C), \tau(D)\} \text{ by (IH);}$$

or $\tau(C\varphi_{i+1}) = \tau(x_{i+1}, M\varphi_i) + \tau(t_{i+1})$
(note that by (IH) - $A(i)-1$
$\tau_{max}(x_{i+1}, M\varphi_i) = \tau_{min}(x_{i+1}, M\varphi_i) = \tau(x_{i+1}, M\varphi_i)$).

But by $SUB(k, P\varphi_i) = t_{i+1}$ it follows
$\tau_{min}(x_{i+1}, M\varphi_i) + \tau(t_{i+1}) \leq \tau(P\varphi_i)$ and in this case
$\tau(x_{i+1}, M\varphi_i) + \tau(t_{i+1}) \leq \tau(P\varphi_i) = \tau(P\varphi_{i+1})$.

Combining these inequalities with (IH) we get

$$\tau(C\varphi_{i+1}) \leq \max\{\tau(C), \tau(D)\}, \ \tau(D\varphi_{i+1}) = \tau(D\varphi_i) \leq \max\{\tau(C), \tau(D)\}.$$

Again the case $SUB(k, M\varphi_i) = t_{i+1}$, $SUB(k, D\varphi_i) = x_{i+1}$ is completely symmetric.
Thus $A(i+1)$ is shown.

Q.E.D.

<u>THEOREM</u> 3.4.1.: Semantic clash resolution with negative setting is a decision procedure for VED.

<u>Proof</u>:
Let $\mathcal{C}$ be an arbitrary set of clauses.
We define $R^0(\mathcal{C}) = \mathcal{C}$

$$R^{i+1}(\mathcal{C}) = R^i(\mathcal{C}) \cup RSC\mathfrak{m}_n(R^i(\mathcal{C})).$$

If $\mathcal{C}$ is a set of Horn clauses then $R^*(\mathcal{C}) - \mathcal{C}$ consists of unit clauses only. In order to show finiteness of $R^*(\mathcal{C})/\sim_v$ it is sufficient to show that for all $C \in R^*(\mathcal{C})$ $\tau(C) \leq \tau(\mathcal{C})$.

We prove by induction on i that $\tau(R^i(\mathcal{C})) \leq \tau(\mathcal{C})$ and $R^i(\mathcal{C}) \in$ VED.

The case $i = 0$ is trivial.

(IH) Suppose that $\tau(R^i(\mathcal{C})) \leq \tau(\mathcal{C})$.

If $C \in R^{i+1}(\mathcal{C}) - R^i(\mathcal{C})$ then C is resolvent of a clash $(E; D_1, \dots, D_n)$ with $E, D_1, \dots, D_n \in R^i(\mathcal{C})$.

By an easy induction on i and by applying lemma 3.4.1 we get $\{E_{i+1}\} \in$ VED and $\tau(E_{i+1}) \leq \tau(\mathcal{C})$ from $\{E, D_{i+1}\} \in$ VED and $\tau(E_i) \leq \tau(\mathcal{C})$

But then $E_n = C$ and thus $\tau(C) \leq \tau(\mathcal{C})$ and $\{C\} \in$ VED.

It follows $\tau(R^{i+1}(\mathcal{C})) \leq \tau(\mathcal{C})$ and $R^{i+1}(\mathcal{C}) \in$ VED.

Therefore $R^*(\mathcal{C}) \in$ VED and $R^*(\mathcal{C})|\sim_v$ is finite.

Q.E.D.

In none of the decision problems studied in section 3.2 we made use of ordering structures for clauses. However, it is well known that in Horn logic fixed ordering of clauses is allowed, making clauses to lists rather than sets. If positive hyperresolution ($RSC\mathfrak{m}_n$) is combined with fixed ordering of rules and goals we come to HOSC - resolution (Horn ordered semantic clash resolution). HOSC - resolution was used in [LG 90] for problems of Horn clause implication, in [Lei90] also for deciding the classes KI, KII mentioned in chapter 3.2. Because set notation for clauses is not very practical in dealing with HOSC - resolution we create some special terminology.

Let $\{P, \neg P_1, \dots, \neg P_n\}$, $\{\neg Q_1, \dots, \neg Q_n\}$ be Horn clauses (a rule and a goal) s.t. there is a total ordering $<_C$ for negative literals in every clause C.

If $C = \{P, \neg P_1, \dots, \neg P_n\}$ and $\neg P_1 < \neg P_2 \; . \; . \; . < \neg P_n$ we write C as $P \leftarrow P_1, \dots, P_n$, what corresponds to the sequent notation usual in logic programming.

We define the HOSC - resolvent of a clash $(C; D_1, \dots, D_n)$ as follows:

$C = P \leftarrow P_1, \dots, P_n$ or $C = \leftarrow P_1, \dots, P_n$ and $D_i = Q_i \leftarrow$ for $i = 1, \dots, n$

(P_i, Q_i are atom formulas).

$R_0 = C$.

Suppose that $R_i = P' \leftarrow P_1', \dots, P_{n-i}'$ where P', P_k' are instances of P, P_k.
If $\{P_{n-i}', D_{i+1}\eta\}$ (for some renaming substitution η) is unifiable by m.g.u. σ

we set

$R_{i+1} = P'\sigma \leftarrow P_1'\sigma, \dots, P_{n-i-1}'\sigma$ for $i < n-1$

$R_{i+1} = P'\sigma \leftarrow$ for $i = n-1$

R_n is called HOSC - resolvent of the clash $(C; D_1, \dots, D_n)$ and is written as $HR(C; D_1, \dots, D_n)$.

Note that, if $C = \leftarrow P_1, \dots, P_n$ (a goal clause), then only $\square$ can be $HR(C; D_1, \dots, D_n)$.
HOSC - resolution is a very efficient refinement because every clash - sequence can produce at most one resolvent; moreover it can be combined with forward subsumption. The set of all HOSC - resolvents definable by clash sequences from $\mathcal{C}$ is denoted by $HRS(\mathcal{C})$.
$HD(\mathcal{C})$ = is the set of all facts which can be derived by HOSC - resolution from $\mathcal{C}$.

More exactly: $H_0 = \text{Facts}(\mathcal{C})$

$$H_{i+1} = H_i \cup HRS(\mathcal{C} \cup H_i) \quad \text{for all } i \text{ and}$$

$$H^* = \bigcup_{i=0}^{\infty} H_i, \quad HD(\mathcal{C}) = H^*|_{\sim_v}$$

Note, that by the recursive definition also $\square$ may be in $HD(\mathcal{C})$.
It was shown in [Lov78] that HOSC - resolution is complete. That means, for every set of Horn clauses $\mathcal{C}$:

$\mathcal{C}$ is unsatisfiable iff $\square \in HD(\mathcal{C})$.

Note that $HD(\mathcal{C})$ is finite if all derivable facts have a common bound on term depth. If in the definition of the H_i only clauses of term depth $\le d$ are allowed, that is

$$H_0' = \{C \mid C \in \text{Facts}(\mathcal{C}),\ \tau(C) \le d\}$$

$$H'_{i+1} = H_i' \cup \{C \mid C \in HRS(\mathcal{C} \cup H_i'),\ \tau(C) \le d\}.$$

We get $H_d(\mathcal{C}) = \bigcup_{i=0}^{\infty} H_i'|_{\sim_v}$
as the set of facts derivable within depth d under equivalence w.r.t. renaming. $H_d(\mathcal{C})$ is always finite and can be constructed algorithmically. Note that $P\leftarrow\ \in H_d(\mathcal{C})$ is a decidable problem (P is an arbitrary atom), while

$P\leftarrow \in \{C \mid C \in HD(\mathcal{C}), \tau(C) \leq d\}$ is not (otherwise, by $\tau(\square) \leq d$ the decision of $\square \in \{C \mid C \in HD(\mathcal{C}), \tau(C) \leq d\}$ would give a decision procedure for the satisfiability problem of Horn logic).

We now turn to the so called clause implication problem:

Let $\forall C$, $\forall D$ be the closed universal formulas corresponding to the clauses C and D. The problem, whether $\forall C \rightarrow \forall D$ is valid, is called the *implication problem for clauses*. In [Sch88] it is proved that this problem is recursively unsolvable. Quite recently, Mycielski and Pacholski proved the unsolvability of the Horn clause implication problem [MP92] (i.e. the problem $\forall C \rightarrow \forall D$, where C is a Horn clause). Because of practical relevance to logic programming and in order to establish sharp borderlines between decidable and undecidable classes, an investigation of decidable subclasses of the Horn clause implication problem is of scientific interest. In [LG90], [Lei90], [Lei88] some decidable Horn classes for the implication problem have been investigated. The decision methods in [Lei88] were either ground methods or methods based on clause powers (iterated self-resolvents), while HOSC-resolution was used in [LG90], [Lei90]. Formally, the Horn clause implication problem is represented by a Horn set $\mathcal{C} = \{C, E_1, \ldots, E_m\}$ where C is a Horn clause and the E_i are either ground facts or ground unit goals (the E_i stem from the Skolemization of $\neg\forall D$). The unsatisfiability of such a set of clauses is equivalent to the validity of the implication problem.

The case that C is a fact or a goal is trivial, because in both cases $\square$ is the only possible clash resolvent. Thus the only interesting case is that C is a rule.

DEFINITION 3.4.2.: A Horn set $\mathcal{C}$ is called a *non-trivial implication problem* (NTI-Problem) if Facts($\mathcal{C}$) and Goals($\mathcal{C}$) consist of ground unit clauses only, $|\text{Rules}(\mathcal{C})| = 1$ and Facts($\mathcal{C}$) $\cup$ Goals($\mathcal{C}$) is consistent.

We now define the class which was called KIII in [Lei90]:

DEFINITION 3.4.3.: The class VH1 (one variable in the head) is the set of all NTI-problems $\mathcal{C}$ where the rule $C \in \mathcal{C}$ is subjected to the restrictions:

a) $|V(C_+)| \leq 1$ and

b) there is a $P \in C_-$ s.t. $V(P) = V(C_-)$.

<u>Remark</u>: If $\mathcal{C}$ is in VH1 and for the rule $C \in \mathcal{C}$ we have $V(C_+) = 0$ then the satisfiability problem for $\mathcal{C}$ is trivial. In fact, if $C_+ = \{P\}$ then $HD(\mathcal{C}) \subseteq \{P \leftarrow, \square\} \cup Facts(\mathcal{C})$.

Hence, the only interesting case is $|V(C_+)| = 1$.
The class VH1, although it seems to be much more "special" than the class PVD, is in fact much harder to handle, the reason of which is that we do not get monotonic behaviour w.r.t. term depth.
Meanwhile the more general case for ≤ 2 variables in the head was solved by V. Rudenko [Rud 91]; his method is also based on HOSC-resolution.

There may be electrons with high term depth which are necessary for the derivation of H-resolvents with low term depth. Fortunately, we can give a bound on such possible "oscillations", and thus get decidability. The following lemma gives a characterization of clauses which are H-derivable from $\mathcal{C}$ for $\mathcal{C} \in$ VH1.

<u>LEMMA</u> 3.4.2.: Let $\mathcal{C}$ be a set of clauses in VH1 and let C be the rule in $\mathcal{C}$ with $C = P \leftarrow P_1, \ldots, P_n$ s.t. $V(P_n) = V(C_-)$. Then either

a) $HD(\mathcal{C})$ is finite or

b) if $E' = HR(C; E_1, \ldots, E_n)$ and $E_1 \in HD(\mathcal{C}) - Facts(\mathcal{C})$ s.t. $E_1 = P\lambda \leftarrow$ for $\lambda = \{s/x\}$ then $E' = P\mu \leftarrow$ where $\mu = \{t/x\}$ for a term t s. t. $\tau(s) < \tau(t)$.

<u>Proof:</u>
It is sufficient to focus on the case $|V(P)| = 1$, because for $|V(P)| = 0$ $HD(\mathcal{C})$ is trivially finite.
Thus suppose that $V(P) = \{x\}$. If $x \notin V(C_-)$ then $P \leftarrow$ is the only possible H-resolvent with nucleus C, and $HD(\mathcal{C})$ is finite.
So we may assume that $x \in V(C_-)$.
We proceed by analyzing $CORR(P,P_n)$:

Case 1) $CORR(P,P_n)$ contains a pair (s,x) for a term s.

Case 1.1) $CORR(P,P_n)$ contains a pair (s,x) s.t. s is a ground term.
Suppose that $Q \leftarrow \in HRS(\mathcal{C})$. Then, because C is the only rule in $\mathcal{C}$ and by $V(P) = \{x\}$, $Q = P\{t/x\}$ for some ground term t.
For abbreviation, we write $P[t]$ instead of $P\{t/x\}$.

Because s is ground, we have $(s,x) \in CORR(P[t],P_n)$. By $V(P_n) = V(C\_)$, every H-resolvent $R\leftarrow$ from a clash $(C; E_1, \dots, E_n)$ is determined by the first electron E_1 only; that means $R = P\lambda$ for λ = m.g.u. of $\{P_n\leftarrow, E_1\}$. On the other hand, E_1 alone cannot guarantee the existence of a clash resolvent; here we have to take into account all of the E_i. Thus the resolvents are in HRS($\mathcal{C}$) if the first electron is in Facts($\mathcal{C}$). Therefore we may suppose that the first electron is of the form $P[t]\leftarrow$ with $P[t]\leftarrow \in HRS(\mathcal{C})$.

So let $E = HR(C; P[t]\leftarrow, E_2, \dots, E_n)$ be a clash resolvent with $P[t]\leftarrow \in HRS(\mathcal{C})$; this clash can only define a resolvent if $\{P[t], P_n\}$ is unifiable.

By $(s,x) \in CORR(P[t],P_n)$, every m.g.u. λ must fulfil $s\lambda = x\lambda$ and by s ground, $s/x \in \lambda$. That means the clash resolvent E is $P[s]\leftarrow$; because $E = P[s]\leftarrow$ for every choice of t, HD($\mathcal{C}$) must be finite.

Case 1.2) For all $(s,x) \in CORR(P,P_n)$, s is not ground.

By $V(P) = \{x\}$ we conclude that s contains x.

If $s = x$ we get $(x,x) \in CORR(P,P_n)$. Let $P[t]\leftarrow \in HRS(\mathcal{C})$ and E be the clash resolvent of $(C; P[t]\leftarrow, E_2, \dots, E_n)$; then $\{P[t], P_n\}$ must be unifiable by a m.g.u. λ.

By $(t,x) \in CORR(P[t],P_n)$ we conclude $t/x \in \lambda$ (note that by $x \in V(C\_)$ t is ground). It follows that $E = P[t]\leftarrow$ where $P[t]\leftarrow \in HRS(\mathcal{C})$ and HD($\mathcal{C}$) is finite.

If $s \neq x$ then $s = g[x]$ for a term $g[x]$ properly containing x. Again, let $E = HR(C; P[t]\leftarrow, E_2, \dots, E_n)$ and λ be m.g.u. of $\{P[t],P_n\}$. By

$(g[x],x) \in CORR(P,P_n)$ we have

$(g[t],x) \in CORR(P[t],P_n)$.

Because $g[t]$ is ground we must have $g[t]/x \in \lambda$ and $E = P[g[t]]$.

By definition of the term $g[x]$ we have $\tau(t) < \tau(g[t])$ and E fulfils condition b).

Case 2) $CORR(P,P_n) \neq \{(P, P_n)\}$, but there is no pair s.t. $(s,x) \in CORR(P,P_n)$.

Because P_n contains x there must be a term $g[x]$ properly containing x and a term s s.t. $(s, g[x]) \in CORR(P,P_n)$ and $(s, g[x])$ is irreducible (in the sense of [MM82]). If s is not a variable then for every renaming substitution η, $(s\eta, g[x])$ is not unifiable. It follows that for any ground instance $P[t]$ $\{P[t], P_n\}$ is not unifiable. So no $E \in HRS(\mathcal{C})$ can be used as first electron in a clash, and $HD(\mathcal{C})$ must be finite.

If $s \in V$ then $s = x$ (by $V(P) = \{x\}$) and $(x, g[x]) \in CORR(P, P_n)$.

Let $g[x] = f(t_1, \dots, t_k)$ for an $f \in FS(\mathcal{C})$ and terms $t_1, \dots, t_k$.

If $E = HR(C;\ P[t] \leftarrow, E_2, \dots, E_n)$ then there is a m.g.u. λ of $\{P[t], P_n\}$.

By $(t, g[x]) \in CORR(P[t], P_n)$ and $g[x] = f(t_1, \dots, t_k)$ we conclude that $t = f(s_1, \dots, s_k)$ for some ground terms s_i (by $x \in V(P_n)$ and $P[t] \leftarrow\ \in HRS(\mathcal{C})$, t must be ground).

Because x must occur in some t_i, λ must contain r/x for a ground term r which is a subterm of an s_j.

It follows that $E = P[r]$ and $\tau(r) < \tau(t)$ and $\tau(E) \le \tau(P[t])$.

Let $d = \max\{\tau(E) \mid E \in HRS(\mathcal{C}) \cup Facts(\mathcal{C})\}$. Then $HD(\mathcal{C}) = HD_d(\mathcal{C})$; but $HD_d(\mathcal{C})$ is finite.

Case 3) $CORR(P, P_n) = \{(P, P_n)\}$.

By $V(P) \subseteq V(P_n)$ the first electron in a clash determines the form of the clash resolvent. But $CORR(P, P_n) = \{(P, P_n)\}$ implies that $\{P, P_n\}$ is not unifiable and no $P[t] \leftarrow\ \in HD(\mathcal{C}) - Facts(\mathcal{C})$ can be used as first electron. It follows that $|HD(\mathcal{C})| \le 2\,|Facts(\mathcal{C})| + 1$.

Combining the parts of the proof we get:

In cases 1.1), 2), 3) $HD(\mathcal{C})$ is finite, while property b) holds in case 1.2).

Q.E.D.

By the proof of lemma 4.1 we also get a method to decide whether a) or b) holds; for such a decision we simply have to analyze $CORR(P, P_n)$. In case a) we compute $HD(\mathcal{C})$. In case b) the situation is more difficult because $HD(\mathcal{C})$ may be infinite. But we will show that there is a bound on term depth d s.t. $\square \in HD(\mathcal{C})$ iff $\square \in HD_d(\mathcal{C})$.

<u>DEFINITION</u> 3.4.4.: Let $\mathcal{C}$ be a set of clauses in VH1 and let C be the rule in $\mathcal{C}$ with $C = P \leftarrow P_1, \dots, P_n$ and $V(P_n) = V(C\_)$.

If $\mathcal{C}$ fulfils condition b) in lemma 4.1 we say that $\mathcal{C}$ is *<u>term depth increasing</u>* (TDI).

<u>LEMMA</u> 3.4.3.: Let $\mathcal{C}$ be TDI and in VH1. Then there is a number d (depending recusively on $\mathcal{C}$) s.t. $\square \in HD(\mathcal{C})$ iff $\square \in HD_d(\mathcal{C})$.

<u>Proof:</u>

Let $C = P \leftarrow P_1,\dots,P_n$ be the single rule in $\mathcal{C}$ and $V(P_n) = V(C_-)$. We exclude the trivial cases $V(P) = 0$ or $V(P) \cap V(C_-) = \emptyset$.

Let $E = HR(C, (E_1,\dots,E_n))$ and $E_1 = Q \leftarrow$. Then E is determined by E_1 only (see also case 1.1 in lemma 4.1).

Obviously, the unifiability of $\{P_n, Q\}$ is a necessary condition for the existence of a resolvent of $(C; E_1,\dots,E_n)$. Thus, if $\{Q, P_n\}$ is unifiable by m.g.u. σ, we call $P\sigma \leftarrow$ a *<u>potential</u>* *<u>H-resolvent</u>* with first electron E_1. The term "potential" is justified because - in case of existence - the resolvent indeed is $P\sigma \leftarrow$.

Let $\Lambda = \{Q \leftarrow \mid Q \leftarrow \in \mathcal{C}$ and $\{Q, P_n\}$ is unifiable$\}$.

Λ is of crucial importance to $HD(\mathcal{C})$ because only facts from Λ, used as first electrons, can lead to successful derivations.

Let $\Lambda = \{Q_1 \leftarrow,\dots,Q_k \leftarrow\}$; then every $PE1(Q_i)$ (potential H-resolvent with first electron Q_i) is of the form $P[s_i] \leftarrow$ where $P[s_i] = P\{s_i/x\}$ and s_i is ground.

If $HRS(\mathcal{C}) = \emptyset$ then $HD(\mathcal{C}) = Facts(\mathcal{C})$ and $HD(\mathcal{C}) = HD_d(\mathcal{C})$ for $d = \tau(\mathcal{C})$.

If $\square \in HRS(\mathcal{C})$ there must exist a fact $Q \leftarrow \in Facts(\mathcal{C})$ and a goal $\leftarrow Q \in goals(\mathcal{C})$, and again we set $d = \tau(\mathcal{C})$ (note that a clash with nucleus C cannot give $\square$).

If $HRS(\mathcal{C}) \neq \emptyset$ and $\square \notin HRS(\mathcal{C})$ we proceed in the following way:

Let $A = \{i \mid P[s_i] \leftarrow\ \in HRS(\mathcal{C})\}$ and $B = \{1,\dots,k\} - A$.

a) $C = P \leftarrow P_1$.

In this case $B = \emptyset$, because P_1 also determines the existence (and not only the form) of the resolvent; so every $PE1(Q_i)$ is an actual resolvent.

Because $\square \notin HRS(\mathcal{C})$, $\square$ can only be in $HD(\mathcal{C})$ if there is a $P[s] \leftarrow\ \in HD(\mathcal{C})$ s.t. $\leftarrow P[s] \in Goals(\mathcal{C})$ (Note that all facts in $HD(\mathcal{C})$ must be ground by $V(P) \subseteq V(P_n)$).

Define $s_{i1} = s_i$ for $i = 1,\dots,k$

$s_{ik+1} = s_{ik}$ if $\{P[s_{ik}], P_n\}$ is not unifiable

$= t$ if $\{P[s_{ik}], P_n\}$ is unifiable by m.g.u. σ and $t = x\sigma$ ($P[t]$ is H-resolvent).

Because $\mathcal{C}$ is TDI we either have $s_{ik+1} = s_{ik}$ and no resolvent can be derived by (single) electron $P[s_{ik}] \leftarrow$ or $\tau(s_{ik+1}) > \tau(s_{ik})$.

If $\tau(P[s_{ij}]) > \tau(Goals(\mathcal{C}))$ then no clause derived from $P[s_{ij}] \leftarrow$ (using $P[s_{ij}] \leftarrow$ as electron in the derivation) can ever resolve with a clause in

Goals($\mathcal{C}$) because Goals($\mathcal{C}$) consists of ground clauses only. Thus we may choose $d = \tau($ Goals($\mathcal{C}$)).

b) $C = P \leftarrow P_1, \dots, P_n$ with $n \geq 2$.

Here we face a situation which is substantially more difficult:

1) $B \neq \emptyset$ is possible and

2) resolvents appearing as n-th electrons for $n > 1$ also determine the existence (but not the form) of further H-resolvents. Let

$\mathcal{C}_0$ = Facts($\mathcal{C}$),

$\mathcal{C}_{k+1}$ = HRS($\mathcal{C}_k \cup \mathcal{C}$) $\cup\, \mathcal{C}_k$ and

$\tau_1 = \max\{ \tau(Q_i) \mid i = 1,\dots,k \}$

By $V(P_n) = V(C\_)$ there is a number r s.t. for all substitutions ϑ $\tau(C\_\vartheta) \leq \tau(P_n\vartheta) + r$.

By the last property we conclude that a clash with first electron $Q_i \leftarrow$ can only be generated if all electrons in the clash are of depth $\leq \tau_1 + r$ ($\tau(C\_\vartheta) \leq \tau_1 + r$ by $P_n\vartheta = Q_i$ for some ϑ); note that for every ϑ with $P_n\vartheta = Q_i$ C_ϑ is ground. Moreover, a contradiction can only be derived if $P' \leftarrow \in$ HD($\mathcal{C}$) resolves with $\leftarrow P'$ in Goals($\mathcal{C}$). For such a P' we have $\tau(P') \leq \tau($ Goals($\mathcal{C}$)).

Again, let us define s_{ij}:

If $P[s_i] \leftarrow \in$ HRS($\mathcal{C}$) (or $i \in A$) then $s_{i1} = s_i$.

Suppose now that $P[s_{ij}] \leftarrow \in \mathcal{C}_j$ and there are $E_2, \dots, E_n \in \mathcal{C}_j$ and a clause E s.t. $E = HR(C; P[s_{ij}] \leftarrow, E_2, \dots, E_n)$ (by definition of the $\mathcal{C}_j$, E is in $\mathcal{C}_{j+1}$).

E must be of the form P[t] for some ground term t. So in the case that E exists we set $s_{ij+1} = t$.

If there is no H-resolvent definable by first electron $P[s_{ij}] \leftarrow$ and facts in $\mathcal{C}_j$ then we set $s_{ij+1} = s_{ij}$.

$$
\begin{aligned}
\text{Define } l_{ij} &= \tau(s_{ij}) \text{ if } P[s_{ij}] \leftarrow \in \mathcal{C}_j \\
&= 0 \text{ if } P[s_{ij}] \leftarrow \notin \mathcal{C}_j \text{ (or } P[s_i] \leftarrow \notin \mathcal{C}_j \text{).}
\end{aligned}
$$

We may represent the "status" of $\mathcal{C}_j$ by the tuple $(l_{1j}, \dots, l_{kj})$.

Suppose that for some $\mathcal{C}_j$ we have 1) $\Box \notin \mathcal{C}_j$ and 2) for all $i = 1,\dots,k$ $l_{ij} = 0$ or $l_{ij} > \tau_2 = \max\{\tau_1 + r, \tau(\mathrm{goals}(\mathcal{C}))\}$.

Then $\Box \notin \mathcal{C}_k$ for all $k \geq j$ and therefore $\Box \notin HD(\mathcal{C})$. This holds because $l_{ik} \geq l_{ij}$ (note that $\mathcal{C}$ is TDI) for all $k \geq j$; so, for $k \geq j$, $P[s_{ik}] \leftarrow$ neither resolves with a goal in $\mathcal{C}$ nor can it contribute to a resolvable clash with first electron Q_i for $i \in B$ (where $l_{ij} = 0$); recall that C_σ is ground for any σ with $\mathrm{dom}(\sigma) = V(P_n)$.
If there is some i s.t. $l_{ij} > 0$ and $l_{ij} \leq \tau_2$ (for a $\mathcal{C}_j$) then $P[s_{ij}] \leftarrow$ (potentially) contributes to new H-resolvents as n-th electron for $n > 1$. If $P[s_{ij}] \leftarrow$ is first electron in a clash, there may be an n-th electron ($n > 1$) of the form $P[s_{kj}] \leftarrow$ s.t. $l_{kj} > \tau_2$, and the clash is resolvable. On the other hand, every clause $P[s] \leftarrow$ s.t. $\tau(P[s]) > \tau(P[s_{ij}]) + r$ is useless in a clash with first electron $P[s_{ij}] \leftarrow$.
No $P[s_{ij}] \leftarrow$ with $l_{ij} \geq \tau_2 + k(r+1)$ can ever contribute to a derivation of $\Box$; this can be explained in the following way:
Let $P[s_{ij}] \leftarrow \in \mathcal{C}_j$ with $l_{ij} \geq \tau_2 + k(r+1)$ and $(m_{1j},\dots,m_{kj})$ be the tupel $(l_{1j},\dots,l_{kj})$ after ordering the components of the last tupel under $\leq$. Then either $m_{ij} > \tau_2$ for all i or there is a maximal number p s.t. $m_{(pj)+1} - m_{pj} > r$; in the first case all relevant clauses are already exhausted and $\Box \in \mathcal{C}_j$ iff there is a $k \geq j$ s.t. $\Box \in \mathcal{C}_k$.
If $m_{(pj)+1} - m_{pj} > r$ s.t. p is maximal w.r.t. this property then every $P[s_{ij}] \leftarrow$ s.t. $l_{ij} \geq m_{(pj)+1}$ is useless as electron in a derivation of a $P[s] \leftarrow$ s.t. $\tau(s) \leq \tau_2$.
Thus it is sufficient to deal with clauses of depth $< \tau_2 + k(r+1)$ only, and by defining $d = \tau_2 + k(r+1)$ we get $\Box \in HD(\mathcal{C})$ iff $\Box \in HD_d(\mathcal{C})$.

Q.E.D.

<u>THEOREM</u> 3. 4. 2.: The satisfiability problem is decidable for the class VH1.

<u>Proof:</u>

Because H-resolution is complete, $\mathcal{C}$ is unsatisfiable iff $\Box \in HD(\mathcal{C})$.
Suppose that $\mathcal{C}$ is in VH1 and that C is the rule in $\mathcal{C}$. If the literal $P' \in C\_$ with $V(P) = V(C\_)$ is not the last literal in $C\_$, we may reorder C to $P \leftarrow P_1,\dots,P_n$

s.t. $P_n = P'$ without loosing completeness. So we may assume that $C = P \leftarrow P_1,\dots,P_n$ and $V(P_n) = V(C_-)$. By the lemmas 3.4.2, and 3.4.3 we conclude that either $HD(\mathcal{C})$ is finite or there is a number d (which can be computed effectively from $\mathcal{C}$) s.t. $\square \in HD(\mathcal{C})$ iff $\square \in HD_d(\mathcal{C})$. Because it is decidable which of both cases applies, we either compute $HD(\mathcal{C})$ or $HD_d(\mathcal{C})$, both being finite. The resulting decision procedure is obvious.

Q.E.D.

Corollary 3.4.1.: Let VH1B2 (B2 for two variables in the body) be the class of all non-trivial implication problems s.t. the rule C fulfils $|V(C_+)| \leq 1$ and $|V(C_-)| \leq 2$. Then VH1B2 is decidable.

Proof:

Let $\mathcal{C} \in$ VH1B2 and C be the rule clause in $\mathcal{C}$.

case a: $|V(C_-)| \leq 1$. Then $V(C_-) = V(P)$ for some $P \in C_-$ and $\mathcal{C} \in$ VH1.

case b: $|V(C_-)| = 2$.

case b_1: There is a $P \in C_-$ s.t. $|V(P)| = 2$. Then $V(P) = V(C_-)$ and - again - $\mathcal{C} \in$ VH1.

case b_2: For all $P \in C_-$ it holds $|V(P)| \leq 1$. Suppose that $V(C_-) = \{x,y\}$,

$K_0 = \{P_1, \dots, P_k\}$ = set of ground atoms in C_- ,
$K_1 = \{Q_1,\dots,Q_l\}$ = set of atoms in C_- containing x,
$K_2 = \{R_1,\dots,R_m\}$ set of atoms in C_- containing y.
Note that $K_1 \cap K_2 = \emptyset$.

Because $|V(C_+)| \leq 1$ we get $V(C_+) \cap V(K_1) = \emptyset$ or $V(C_+) \cap V(K_2) = \emptyset$.

Let w.l.o.g. $V(C_+) \cap V(K_2) = \emptyset$.
Then $V(C_+ \cup K_0 \cup K_1) \cap V(K_2) = \emptyset$ and C can be decomposed.
More exactly, let $C_1 = P \leftarrow P_1,\dots,P_k,Q_1,\dots,Q_l$ (for $C_+ = \{P\}$), $C_2 = \leftarrow R_1,\dots,R_m$ and

$$\mathcal{C}_1 = (\mathcal{C} - \{C\}) \cup \{C_1\},\quad \mathcal{C}_2 = (\mathcal{C} - \{C\}) \cup \{C_2\}.$$

Obviously, $\mathcal{C}$ is unsatisfiable iff both $\mathcal{C}_1$ and $\mathcal{C}_2$ are unsatisfiable. But $\mathcal{C}_1$ is in VH1 and $\mathcal{C}_2$ is a trivial implication problem.

Q.E.D.

For deciding VH1 we didn't use full HOSC-resolution, but a modification which cuts terms at some depth d, which is computed from the structure of the implication problem. While this modification always terminates, it is not complete. The problem was that only variable occurrences, but not the term structure itself was restricted in VH1. To compute $HD_d(\mathcal{C})$ for some depth d is much more a "classical" decision algorithm than a theorem prover. The method was refined by V. Rudenko [Rud 91] by defining other complexity measures for atoms than term depth; by this method he succeeded to show, for the case $|V(C_+)| \leq 2$, $V(C) = V(L)$ for a $L \in C_-$, that either $HD(\mathcal{C})$ is finite or $HD(\mathcal{C})$ defines a monotonically increasing chain (w.r.t complexity) of facts; thus he obtained an improved decision algorithm for deciding Horn implication problems. His method shows that HOSC-resolution is a good basis structure to design decision algorithms for such problems. It seems to be that the problem gets substantially more difficult if only $|V(C_+)| \leq 2$ is stated in the definition, without restricting variable occurrence in C_-. The more general case, $V(L) = V(C)$ for an $L \in C_-$, without restriction on the number of variables in the head, is still unsolved.