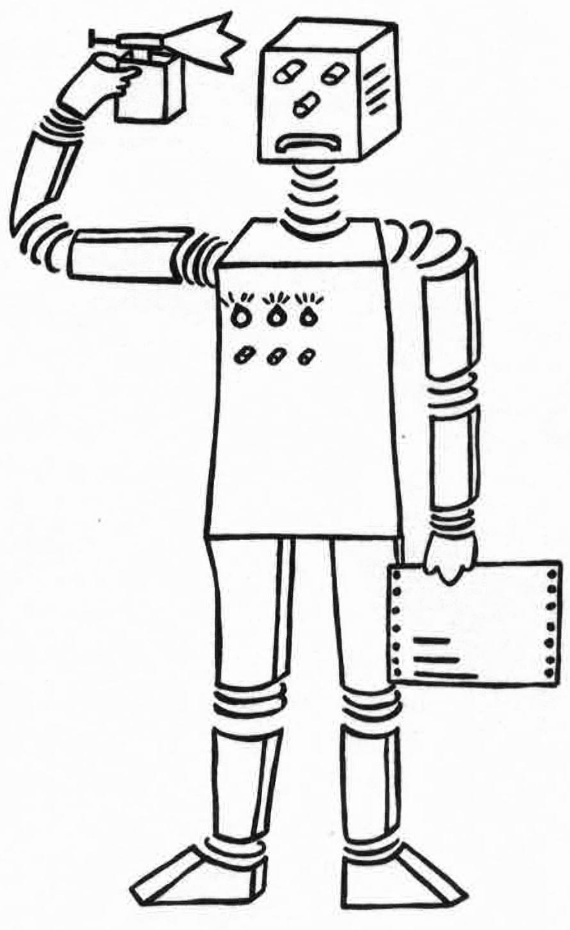
SEKI-REPORT

Artificial
Intelligence
Laboratories

Fachbereich Informatik
Universität Kaiserslautern
Postfach 3049
D-6750 Kaiserslautern 1, W. Germany



Using Theory Resolution to Simplify Interpreted Formulae

Rolf Socher-Ambrosius

SEKI Report SR-88-16

Using Theory Resolution to Simplify Interpreted Formulae

Rolf Socher-Ambrosius
Fachbereich Informatik, Universität Kaiserslautern
Postfach 3049, D-6750 Kaiserslautern, W.-Germany

Abstract: Loveland & Shostak (1980) gave an algorithm for converting a decision procedure for ground formulae in a first-order theory to a simplifier for such formulae. The algorithm is supposed to produce a simplest clause set among all formulae built from atoms of the original formula. In this paper it is shown that this method does not meet this requirement. Furthermore we show that theory resolution can be used to extend Quine's method to an algorithm that really accomplishes the task of simplifying interpreted formulae.

1 Introduction

The problem of simplifying logical formulae originated in the 1950s in connection with the problem of minimizing the number of components of a given switching circuit. Similar problems arose later in the areas of program verification and automated deduction. However, in these domains not only expressions of propositional logic are to be simplified, but also formulae of the first order predicate logic. In this paper we consider the problem of the representation of propositional formulae as minimal clause sets; the transformation of our results to the analogous situation with disjunctive normal form (which was used by Loveland & Shostak) is obvious.

There is a well-known technique to minimize expressions of propositional logic, which was developed by Quine (1952) and (1959). This method, the *method of iterated consensus*, when applied to a propositional formula $\mathcal{F}$ in clausal normal form, results in a set $\mathcal{P}_{\mathcal{F}}$ of prime implicants. These are the clauses, which are minimal (with respect to the subsumption order) among those implied by $\mathcal{F}$. This means that each clause that is implied by $\mathcal{F}$, either is a member of $\mathcal{P}_{\mathcal{F}}$ or is subsumed by some member of $\mathcal{P}_{\mathcal{F}}$. In general the set of prime implicants of a formula $\mathcal{F}$ still contains redundant clauses and therefore can be reduced to some "simplest equivalent" of $\mathcal{F}$. This is a minimal subset of $\mathcal{P}_{\mathcal{F}}$, which is still logically equivalent to $\mathcal{F}$. But while the set of prime implicants of $\mathcal{F}$ is uniquely determined, the simplest equivalent is not.

<u>1.1 Example</u>

Consider the formula $\mathcal{F} \equiv (p \Leftrightarrow q) \wedge (q \Leftrightarrow r)$. It is easy to see that the set $\mathcal{P} = \{\neg p \vee q, p \vee \neg q, \neg p \vee r, p \vee \neg r, \neg q \vee r, q \vee \neg r\}$ is the set of prime implicants of $\mathcal{F}$. But the following subsets $\{\neg p \vee q, \neg q \vee r, p \vee \neg r\}$, $\{p \vee \neg q, q \vee \neg r, \neg p \vee r\}$ and $\{\neg p \vee q, p \vee \neg q, \neg p \vee r, p \vee \neg r\}$ of $\mathcal{P}$ are all simplest equivalents to $\mathcal{F}$.

The method of iterated consensus consists in the successive formation of resolvents and the removal of subsumed clauses. This yields the set of prime implicants of the original formula. A simplest equivalent is obtained by a selection of an appropriate subset of the prime implicants.

This method has been modified by Loveland & Shostak (1980) in order to simplify interpreted ground formulae. These formulae contain interpreted symbols that belong to some theory with known decision procedure. According to this method, the formula is first transformed into a clause set S over some set A of atoms. Next, the decision procedure for the theory is used to test all $2^{|A|}$ clauses that are composed solely of atoms of A, either negated or unnegated, on validity in the underlying theory. The valid ones, the so called "don't-care" conditions, are added to the original clause set S. This extended clause set is subjected to the methods of iterated consensus and selection of a simplest equivalent.

The following example is taken from Loveland & Shostak (1980).

<u>1.2 Example</u>

Let $\mathcal{T}$ be the theory of Presburger arithmetic (i.e. the theory of real numbers together with addition, linear multiplication and the usual ordering relations).
Assume the formula $\mathcal{F} \equiv a \leq b \wedge (c > -3/2 \vee a+2c-b > -3)$ is to be simplified. If we let p,q,r denote the atoms $a \leq b,\ c > -3/2,\ a+2c-b > -3$, respectively, $\mathcal{F}$ can be written $p \wedge (q \vee r)$, or as a clause set $S=\{p,qr\}$.
Now all clauses $pqr,\ pq\neg r, p\neg qr,\ldots,\neg p\neg q\neg r$ are subjected to a decision procedure for the theory $\mathcal{T}$, like the one developed by Shostak (1981). It turns out that only the clauses $\neg pq\neg r$ and $p\neg qr$ are valid. This yields the extended clause set $S^*=\{p,\ qr,\ \neg pq\neg r, p\neg qr\}$. The method of iterated consensus now yields the resolvents $q\neg r$ and q. The unit clauses p and q subsume all other clauses from which we obtain the set of prime implicants $\mathcal{P} = \{p,q\}$. Moreover, as no proper subset of $\mathcal{P}$ is semantically equivalent to the original formula, $\mathcal{P}$ is already a simplest equivalent.

This method suffers from two serious shortcomings: A technical one and another, more conceptual one. The technical drawback consists in the requirement that 2^n clauses must be submitted to the decision procedure, where n is the number of different atoms of $\mathcal{F}$. However, the method can be improved by the observation, that it is sufficient to test those clauses that are not already subsumed by clauses of the original clause set. In our example only the three clauses $\neg p\neg qr$, $\neg pq\neg r$ and $\neg p\neg q\neg r$ remain to be submitted to the decision procedure. The other deficiency of the method is its lack of satisfying results, as the following example shows:

<u>1.3 Example:</u>

a) Let $\mathcal{T}$ be the theory of Presburger arithmetic and let $S = \{a\geq 0,\ a\geq 1\}$. We try to minimize S according to the LS-method. Let $p,\ q$ denote the atoms $a\geq 0$ and $a\geq 1$, respectively. The clause $p\neg q$ is the only combination of atoms of S that is valid under $\mathcal{T}$. But this clause is subsumed by the clause p, hence it need not be added to the set S. Therefore the extended clause set is the original one and, as no resolution steps can be accomplished, the result of the LS-method is the original set. However, the clause $a\geq 0$ is obviously implied by $a\geq 1$ in the theory $\mathcal{T}$. Hence the set $\{a \geq 1\}$ is the desired minimal equivalent, which points out that the LS-approach does not yield the intended result.

b) Let $\mathcal{T}$ be the theory of equality with uninterpreted function symbols and let $S = \{ffx=x, fffx=x\}$. Again application of the LS-method results in the original set S. However, the set $\{fx=x\}$ is obviously equivalent to S in the theory, which again shows that there may exist simpler clause sets than the LS method produces.

This paper presents a method to simplify interpreted formulae that overcomes these deficiencies. Our method proceeds from the original clause set (without adding don't care conditions) and generates successively *theory resolvents* similarly to the method of iterated consensus. In addition, subsumed clauses (with respect to the underlying theory) are removed. The resulting set is the set of "theory prime implicants". In this paper it will be shown that, in analogy to the uninterpreted case, the prime implicants of an interpreted formula $\mathcal{F}$ are the minimal clauses in the deductive closure of $\mathcal{F}$. Here, "minimal" is to be understood with respect to implication under the given theory. From this follows that the set of theory prime implicants is uniquely determined by $\mathcal{F}$. Again as in the uninterpreted case the set of prime implicants of a formula $\mathcal{F}$ can be further reduced to some "simplest equivalent" of $\mathcal{F}$, which is not uniquely determined.

Theory resolution, first proposed by M. Stickel (1985), is a generalization of ordinary resolution. It is based on the following idea: The literals resolved upon may not be syntactically complementary, it is sufficient that they are complementary in some theory. Take for instance the clauses $C = a>b \vee C'$ and $D = a \leq b \vee D'$. As $a>b$ and $a \leq b$ are complementary in the theory of a partial order, C and D can be resolved giving $C' \vee D'$. Resolvents of more than two clauses can be formed, too. An even more generalized kind of resolution is *partial* theory resolution, including total theory resolution as a special case. Suppose the literals to be resolved upon are complementary under some condition. Then the resolvent can be formed as above with the negated condition (the so called *residue*) added to it. For instance let $C = a>b \vee C'$ and $D = b>c \vee D'$. The literals $a>b$ and $b>c$ are complementary, provided that the condition $c \geq a$ holds. Thus $c<a$ is a residue of the two literals and there is a partial theory resolution step giving $c<a \vee C' \vee D'$.

We show by means of example 1.2 how the method to simplify interpreted formulae works:

1.4 Example

We have $\mathcal{F} \equiv a \leq b \wedge (c > -3/2 \vee a+2c-b > -3)$, written as a clause set $S = \{p, qr\}$. The literals $a \leq b$ and $a+2c-b > -3$ are complementary under the condition $c \leq -3/2$. Hence $c > -3/2$ is a residue of p and r and the appropriate theory resolution step yields the resolvent $c > -3/2 \vee c > -3/2$, which simplifies to $c > -3/2$. This resolvent subsumes its parent clause $c > -3/2 \vee a+2c-b > -3$. Removing this clause we obtain $a \leq b$ and $c > -3/2$ as the only theory prime implicants of $\mathcal{F}$.

2 Theory Resolution

We will assume the standard definitions of a *term*, an *atomic formula* and a *literal.* We will consider a *clause* to be a set of literals. All clauses are *ground clauses,* i.e. they do not contain variables. We just review the basic notions of theory resolution as it is defined by Stickel (1985), see also Bläsius & Bürckert (1987).

In the following $\mathcal{T}$ is a theory which admits a decision procedure. A **$\mathcal{T}$-interpretation** is an interpretation that satisfies theory $\mathcal{T}$. A set of clauses S is **$\mathcal{T}$-unsatisfiable**, iff no $\mathcal{T}$-interpretation satisfies S. S is **minimally $\mathcal{T}$-unsatisfiable**, iff S but no proper subset of S is $\mathcal{T}$-unsatisfiable. The **semantic closure** of S is the set of all formulae $\mathcal{F}$ such that $S \cup \{\neg \mathcal{F}\}$ is $\mathcal{T}$-unsatisfiable.

2.1 Definition:

Let $C_1,..,C_n$ be nonempty clauses, let each C_i be decomposed as $K_i \vee L_i$ with unit clause K_i and let $R_1,...,R_m$ $(m \geq 0)$ be unit clauses. Suppose the set of clauses $K_1,...,K_n,R_1,...,R_m$ is (minimally) $\mathcal{T}$-unsatisfiable. Then the clause $L_1 \vee ... \vee L_n \vee \neg R_1 \vee ... \vee \neg R_m$ is a **(partial) $\mathcal{T}$-resolvent** of $C_1,...,C_n$. The clause $\neg R_1 \vee ... \vee \neg R_m$ is called the **residue** of the theory resolution operation. If the residue is the empty clause, then the $\mathcal{T}$-resolvent is **total.**

In order to guarantee the completeness of the procedure, we require the set of clauses $K_1,...,K_n$, $R_1,...,R_m$ to be minimally $\mathcal{T}$-unsatisfiable. In this case the residue is the strongest consequence of the set $K_1,...,K_n$ and we call it a most general residue. But there may still be more than one most general residue, an example is provided in Stickel (1985). For our purposes we assume a theory $\mathcal{T}$ that admits only a finite set of most general residues.

Stickel (1985) distinguishes between *wide* theory resolution, which allows the clauses K_i to be arbitrary clauses and *narrow* theory resolution, where the K_i are unit clauses. Since only narrow theory resolution is considered in this paper, we simply speak of theory resolution. Also the exclusive use of partial theory resolution accounts for the omission of the word *partial* in this paper.

Soundness and completeness of theory resolution are given by the following theorems (Stickel 1985).

2.2 Theorem:

Let $\mathcal{T}$ be a theory, S a set of clauses and C a $\mathcal{T}$-resolvent of S. Then every $\mathcal{T}$-interpretation $\mathfrak{I}$ that satisfies S also satisfies C. ■

2.3 Theorem:

Let S be a $\mathcal{T}$-unsatisfiable set of clauses. Then there is a refutation of S using (partial or total) theory resolution with theory $\mathcal{T}$ ■

3 Simplifying Interpreted Formulae

In this section we first define some basic notions of the interpreted propositional logic. Then we describe the appropriate modification of the method of iterated consensus and prove that it produces the set of theory prime implicants of the original formula. As the underlying theory is always $\mathcal{T}$, we omit the index $\mathcal{T}$ in some notions.

3.1 Definition (semantic notions):

Let C and D be clauses, $\mathcal{T}$ a theory.

(i) C **$\mathcal{T}$-implies** D, iff $C \wedge \neg D$ is $\mathcal{T}$-unsatisfiable, which is denoted by $C \leq_{\mathcal{T}} D$.

(ii) C and D are called **$\mathcal{T}$-equivalent** (or **semantically equivalent**), iff C $\mathcal{T}$-implies D and D $\mathcal{T}$-implies C, which is denoted by $C \equiv_{\mathcal{T}} D$.

3.2 Definition (syntactic notions):

Let C and D be clauses, $\mathcal{T}$ a theory.

(i) C **$\mathcal{T}$-subsumes** D, iff for each literal L of C there exists a literal K of D such that $L \wedge \neg K$ is $\mathcal{T}$-unsatisfiable, which can be recognized with the decision procedure for $\mathcal{T}$.

(ii) $C' \subset C$ is called a **$\mathcal{T}$-factor** of C, iff for each literal $L \in C \setminus C'$ there is some literal $K \in C'$ such that L $\mathcal{T}$-subsumes K.

(iii) C is called **tautological**, iff $\neg C$ is $\mathcal{T}$-unsatisfiable.

It is easy to see that $\equiv_{\mathcal{T}}$ is an equivalence relation and $\leq_{\mathcal{T}}$ is a partial order on $\mathcal{C}/\equiv_{\mathcal{T}}$, the set of all clauses modulo $\mathcal{T}$-equivalence. In the following we always consider $\mathcal{C}/\equiv_{\mathcal{T}}$, i.e. we assume semantically equivalent clauses to be identical.

In analogy to the uninterpreted case, the connection between $\mathcal{T}$-implication and $\mathcal{T}$-subsumption is given by the following

3.3 Lemma:

Let C and D be nontautological clauses. Then C $\mathcal{T}$-implies D, iff C $\mathcal{T}$-subsumes D.

Proof: Gottlob (1987) proved the following theorem for the predicate logic: If C is not self-resolving and D is not tautological, then subsumption is equivalent to implication. This proof generalizes to total theory resolution. As we deal only with ground clauses, our result follows from the fact that nontautological ground clauses cannot be self-resolving. ■

3.4 Definition:

Let S be a set of clauses. A clause, which is minimal (w.r.t. the $\mathcal{T}$-implication order) in the semantic closure of S is called a **$\mathcal{T}$-prime implicant** of S. The set of $\mathcal{T}$-prime implicants of S is denoted by $\mathcal{P}_S$.

The modified *method of iterated consensus* now works as follows: Let S be a set of clauses. The nontautological $\mathcal{T}$-resolvents of clauses in S are formed and added to the set. Each clause possessing a factor is replaced by this factor. At the same time, $\mathcal{T}$-subsumed clauses are deleted. When no new clauses can be added that are not $\mathcal{T}$-subsumed by existing clauses, the algorithm terminates.

Before proving the main theorem about the correctness and termination of this method, we give a lemma, which is the interpreted counterpart of a well-known lemma of the uninterpreted propositional logic. However, as partial theory resolution is involved, the proof is not fully analogous to the uninterpreted case.

3.5 Lemma:

Let S be a set of clauses. For each nontautological clause C in the semantic closure of S, there exists a clause D with $D \leq_{\mathcal{T}} C$ and a deduction of D from S using $\mathcal{T}$-resolution (not necessarily total).

Proof: If C is in the semantic closure of S, then $S \cup \{\neg C\}$ is $\mathcal{T}$-unsatisfiable and, according to the completeness of $\mathcal{T}$-resolution, admits a $\mathcal{T}$-refutation (i.e. a $\mathcal{T}$-resolution deduction of the empty clause). Let $\neg C = K_1 \wedge \ldots \wedge K_n$. Let S' be the set resulting from S by deleting all clauses that contain a literal which is $\mathcal{T}$-subsumed by some K_i and by removing from the remaining clauses those literals that $\mathcal{T}$-subsume some $\neg K_j$. $S' \cup \{\neg C\}$ is still $\mathcal{T}$-unsatisfiable, hence admits a refutation using total narrow $\mathcal{T}$-resolution. We transform each total resolution step s of this refutation into a partial $\mathcal{T}$-resolution step p in the following way: if s uses the clauses $C_1,..,C_n$, $K'_1,.., K'_m$, with $K'_i \in \{K_1,..,K_n\}$, and produces the resolvent R, then p is the step deriving the resolvent $R \vee \neg K'_1 \vee \ldots \vee \neg K'_m$ from the clauses $C_1,..,C_n$. Then we obtain a partial $\mathcal{T}$-resolution deduction, starting from S' and resulting in a clause C' that is a subset of C. Finally, we restore those original clauses that contained a literal subsuming some $\neg K_j$. This is done by adding those literals to the appropriate clauses in the whole deduction. Let C'' be the set of all these clauses. Now we have obtained a $\mathcal{T}$-deduction of a clause $D := C' \cup C''$ from the original set S. It is easy to see that $\{D, \neg C\}$ is $\mathcal{T}$-unsatisfiable, i.e. $D \leq_{\mathcal{T}} C$. ■

Example 1.2 shows that lemma 3.5 does not hold for total theory resolution. The clause $c > -3/2$ is in the semantic closure of S, but S admits no total theory resolution step at all.

3.6 Theorem:

Let S be a clause set and $\mathcal{P}$ be a set of clauses generated by the method of iterated consensus w.r.t. the theory $\mathcal{T}$. Then $\mathcal{P}$ is the set of $\mathcal{T}$-prime implicants of S.

Proof: Let $\{R_i \mid i \in N\}$ be the set of theory resolvents of clauses of S. According to lemma 3.5, for each nontautological clause C in the semantic closure of S there is some R_j with $R_j \leq_{\mathcal{T}} C$. Let $\mathcal{L}$ be the set of all literals of S and let $\mathfrak{R}$ be the finite set of all possible most general residues of literals from $\mathcal{L}$. Furthermore, let $L_{\mathfrak{R}}$ be the literal set of $\mathfrak{R}$. Then the clauses R_i are clauses over the finite literal set $\mathcal{L} \cup L_{\mathfrak{R}}$, hence the set $\{R_i \mid i \in N\}$ is finite and the termination of the procedure is assured. It remains to show the minimality of $\mathcal{P}$. From lemma 3.3 follows that it is sufficient to remove $\mathcal{T}$-subsumed clauses, instead of $\mathcal{T}$-implicated ones. This removal of subsumed clauses guarantees the desired minimality of the clause set generated by the method of iterated consensus. ■

In order to characterize the notion of simplest equivalent, the prime implicants are classified in three categories:

3.7 Definition:

Let S be a clause set and $\mathcal{P}$ the set of prime implicants of S.

(i) $P\in\mathcal{P}$ is a **core implicant**, iff $\mathcal{P}\setminus\{P\}$ does not imply P, i.e. $\mathcal{P}\setminus\{P\}\cup\{\neg P\}$ is $\mathcal{T}$-satisfiable.

(ii) $P\in\mathcal{P}$ is an **absolutely eliminable implicant**, iff the set of core implicants implies P.

(iii) $P\in\mathcal{P}$ is an **eliminable implicant**, iff it is neither core nor absolutely eliminable.

This definition includes the original definition for the uninterpreted case, i.e. for the empty theory. Any simplest equivalent consists of the set of core implicants and an appropriate subset of the eliminable implicants. The core implicants can be determined with theory resolution.

3.8 Example:

This example is taken again from the theory $\mathcal{T}$ of Presburger Arithmetic.
Let $S = \{C_1, C_2, C_3\}$ with $C_1 = a>0 \vee b>0$, $C_2 = b>a \vee a>c$ and $C_3 = c>1$. C_2 and C_3 can be resolved giving $C_4 = b>a \vee a>1$, which resolves with C_1 to $C_5 = b>0 \vee a>1$. This resolvent subsumes its parent clause C_1, which can therefore be deleted. Having obtained the set $\{C_2, C_3, C_4, C_5\}$, all possible steps lead to resolvents that are subsumed by existing ones. Hence this set respresents the prime implicants of S. From all resolvents of clauses of $\mathcal{P}_S$ only the resolvent C_4 of C_2 and C_3 is not subsumed by some element of $\mathcal{P}_S$. Hence none of the clauses C_2, C_3, C_5 can be deduced from the rest, in other words these clauses are core implicants. As C_4 is implied by the two core implicants C_2 and C_3, it is absolutely eliminable. Thus the set $\{C_2, C_3, C_5\}$ is a simplest equivalent to S.

The following lemma sets up the relation between our method and Loveland & Shostak's method. It shows that the two methods coincide, if our method is restricted to residues appearing as atoms in the original clause set, and if the LS-method is extended by taking into account $\mathcal{T}$-subsumption instead of usual subsumption.

3.9 Lemma:

Let S be a clause set and $\mathcal{P}$ the subset of $\mathcal{T}$-primeimplicants of S that consist only of atoms of S. Let Q be the result of the LS-method applied to S. Then $\mathcal{P}\subseteq Q$.

Proof: Let IC be the method of iterated consensus, as described above. Let IC' be the variant of IC that takes into account only those residues that are atoms of S. Then IC' obviously produces the set $\mathcal{P}$, when applied to S. We show that any resolvent produced by IC' is also generated by LS:

Let C = P$\vee$C' and D=Q$\vee$D' be clauses of S, and suppose R is a residue of P and Q that is also an atom of S. Then the $\mathcal{T}$-resolvent R$\vee$C'$\vee$D' is derived by IC'. As R is a residue of P and Q, the clause $\neg P \vee \neg Q \vee R$ must be valid in $\mathcal{T}$, and is therefore added to S by LS. This enables LS to derive the following resolvents: $\neg Q \vee R \vee C'$, $\neg P \vee R \vee D'$, $R \vee C' \vee D'$. Thus the $\mathcal{T}$-resolvent $R \vee C' \vee D'$ is also derived by LS. The other clauses produced by LS are $\mathcal{T}$-subsumed by C and D, respectively.

Even in the case, where the two methods coincide, our approach appears to be preferable: First, as the proof of the previous lemma showed, the LS method in general produces much more $\mathcal{T}$-subsumed clauses. Second, there are more clauses that have to be submitted to the decision procedure: The IC' method takes n literals $L_1,..,L_n$ from different clauses $C_1,..,C_n$ and looks for an atom A of S being a residue of $L_1,..,L_n$. This amounts to testing (Horn) clauses of the form $\neg L_1 \vee ... \vee \neg L_n \vee A$ on validity in the theory. The LS-method, on the other hand, has to submit not only those Horn clauses, but all different distributions of the negation sign on this clause, such as $\neg L_1 \vee ... \vee \neg L_n \vee \neg A$, $L_1 \vee ... \vee L_n \vee A$, and so on. Moreover, also literals of the same clause may appear among the L_i.

References

Bläsius, K.H. & Bürckert, H.-J. (Eds.) (1987). Deduktionssysteme. Oldenbourg Verlag, München.

Gottlob, G. (1987). Subsumption and Implication. *Information Processing Letters* **24**, 109 - 111.

Loveland, D.W. & Shostak, R.E. (1980). Simplifying Interpreted Formulas. In: W. Bibel & R. Kowalski (Eds.): *Proc. of 5th Conference on Automated Deduction, Les Arcs.* Springer LNCS 87, 97 - 109.

Quine, W.V. (1952). The Problem of Simplifying Truth Functions. *American Math. Monthly*, **59**, 521 - 531.

Quine, W.V. (1959). On Cores and Prime Implicants of Truth Functions. *American Math. Monthly*, **66**, 755 - 760.

Shostak,R.E. (1981). Deciding Linear Inequalities by Computing Loop Residues. *Journal of the ACM*, **28/4**, 769 - 779.

Stickel, M. E. (1985). Automated Deduction by Theory Resolution. *Journal of Automated Reasoning*, **1/4**, 333 - 356.