UPPSALA UNIVERSITET
Matematiska institutionen
Erik Palmgren

Kompletterande material
Tillämpad logik DV1
2002-10-01

The following are some supplementary notes to Chapter 3 and 4 of Das (1992).

1 Some logical background to the resolution method

The resolution method is the basis for many of the most successful automatic theorem provers, e.g. Otter, and logic programming, e.g. PROLOG. It was introduced in 1965 by Alan Robinson.

Definition 1.1 A *literal* is a formula of the form P or $\neg P$ where P is *atomic.* Here P is called a *positive,* while $\neg P$ is *negative.* Two literals L and M are *complementary* if $L = \neg M$ or $M = \neg L$.

Example 1.2 P, $\neg P$, $R(x, f(x))$, $\neg R(g(y), x)$ are literals.

A formula of the form

$$\bigwedge_{i=1} C_i$$

where $C_i = L_{i,1} \vee \cdots \vee L_{i,m_i}$ and each $L_{i,j}$ is a literal, is said to be on *conjuctive normal form*(CNF). (It is *k-CNF* if $m_i \leq k$ for all i.)

Recall that every quantifier free formula is equivalent to a CNF-formula.

Exercise: Is the propositional formula $P \vee Q \vee R$ is logically equivalent to some 2-CNF formula?

A formula is on *Prenex normal form* (PNF) if it has the form

$$Q_1 x_1 Q_2 x_2 \cdots Q_n x_n \; \varphi \tag{1}$$

where $Q_i \in \{\forall, \exists\}$ are quantifiers and φ is quantifier free.

Every formula is logically equivalent to some PNF formula.

Resolution theorem proving depends on the existence of another more restrictive normal form. A closed formula is on *Skolem normal form* (SNF) if it is of the form

$$\forall x_1 \forall x_2 \cdots \forall x_n \; \psi \tag{2}$$

where ψ is on CNF. A *clause* is a closed formula of the shape

$$\forall x_1 \forall x_2 \cdots \forall x_m \, (L_1 \vee \cdots \vee L_n) \tag{3}$$

where each L_i is a literal. In case there is only one literal ($n = 1$), it is usually called a *unit clause.* By convention, a disjunction is absurdity $\perp$ when $n = 0$. The clause is then called *the empty clause* and is denoted $\Box$.

In propositional logic, any disjunction of literals is called a *clause.*

The following proposition is easily proved using the Prenex operations.

Proposition 1 Any formula on Skolem normal form is logically equivalent to a conjunction of clauses. $\Box$

Because of this result a SNF formula can be treated as a set of clauses. It customary to leave the universal quantifiers implicit. Thus

$$\{P(x) \vee \neg R(x), R(x) \vee Q(y)\}$$

stands for the conjunction of clauses

$$\forall x \, (P(x) \vee \neg R(x)) \wedge \forall xy \, (R(x) \vee Q(y)).$$

1.1 Skolemization

Let $\mathcal{A}$ be an L-structure. Then

$$\mathcal{A} \models \forall x_1 \cdots x_n \, \exists y \, \varphi(x_1, \ldots, x_n, y) \tag{4}$$

implies that there is some function $f : A^n \to A$ such that for all $\bar{a} = (a_1, \ldots, a_n) \in A^n$

$$\mathcal{A} \models \varphi(\bar{a}, f(\bar{a}))$$

We extend the language L to L' with a new function symbol F. Expand $\mathcal{A}$ to $\mathcal{A}'$ with $F^{\mathcal{A}'} = f$. Then the above says

$$\mathcal{A}' \models \forall x_1 \cdots x_n \varphi(x_1, \ldots, x_n, F(x_1, \ldots, x_n)). \tag{5}$$

Since (4) and (5) are both true, we have an expanded model $\mathcal{A}'$ in which

$$\mathcal{A}' \models \forall x_1 \cdots x_n \, \exists y \, \varphi(x_1, \ldots, x_n, y) \leftrightarrow \forall x_1 \cdots x_n \, \varphi(x_1, \ldots, x_n, F(x_1, \ldots, x_n)). \tag{6}$$

Now suppose that (4) is false. Then there exists some $\bar{a} \in A^n$ such that for all $b \in A$

$$\mathcal{A} \not\models \varphi(\bar{a}, b).$$

Thus for *any function* $f : A^n \to A$ and $F^{\mathcal{A}'} = f$ we have

$$\mathcal{A} \not\models \forall x_1 \cdots x_n \, \varphi(x_1, \ldots, x_n, F(x_1, \ldots, x_n)).$$

Consequently both sides of (6) becomes false, and hence (6) is true for any choice of f.

We conclude that for any $\mathcal{A}$, and regardless of (4) being true or false, there is some expanded structure $\mathcal{A}'$ such that (6) is true.

Consider a PNF-formula $\varphi = Q_1 x_1 Q_2 x_2 \cdots Q_n x_n \;\; \psi(x_1, \ldots, x_n)$. For each existential quantifier $Q_i = \exists$, remove $\exists x_i$ and replace x_i by $f_i(x_{j_1}, \ldots, x_{j_m})$ where f_i is a new m-ary function symbol and $x_{j_1}, \ldots, x_{j_m}$ are the universally quantified variables to the left $Q_i x_i$. The resulting formula φ^S is called the *Skolemization* of φ, and the new function symbols are called *Skolem function symbols.*

Example 1.3 Here is an example of Skolemization

$$\begin{array}{rcll}
\varphi & = & \underline{\exists x_1} \, \forall x_2 \, \forall x_3 \, \exists x_4 \, R(x_1, x_2, x_3, x_4) & \\
& \mapsto & \forall x_2 \, \forall x_3 \, \exists x_4 \, R(f_1, x_2, x_3, x_4) & f_1 \text{ new constant symbol} \\
& = & \forall x_2 \, \forall x_3 \, \underline{\exists x_4} \, R(f_1, x_2, x_3, x_4) & \\
& \mapsto & \forall x_2 \, \forall x_3 \, R(f_1, x_2, x_3, f_4(x_2, x_3)) & f_4 \text{ new function symbol} \\
& = & \varphi^S &
\end{array}$$

The next lemma gives the relation of a formula and its Skolemization.

Lemma 1.4 *Let φ be a closed prenex L-formula. Let φ^S be its Skolemization and $f_1, \ldots, f_m$ the introduced Skolem function symbols. Then for every L-structure $\mathcal{A}$ there is an $L' = L \cup \{f_1, \ldots, f_m\}$-structure $\mathcal{B}$ expanding $\mathcal{A}$ ($\mathcal{B}|L = \mathcal{A}$) so that*

$$\mathcal{B} \models \varphi \leftrightarrow \varphi^S.$$

Proof. The proof goes by succesively applying the procedure above (4–6) to the outmost existential quantifier, and then noting that an equivalence holds in a structure if it already holds in a restriction. We take the above example again, and find successively f_1 and f_4

$$\begin{array}{rcl}
(\mathcal{A}; f_1) & \models & \exists x_1 \forall x_2 \, \forall x_3 \, \exists x_4 \, R(x_1, x_2, x_3, x_4) \leftrightarrow \forall x_2 \, \forall x_3 \, \exists x_4 \, R(f_1, x_2, x_3, x_4) \\
(\mathcal{A}; f_1, f_4) & \models & \forall x_2 \, \forall x_3 \, \exists x_4 \, R(f_1, x_2, x_3, x_4) \leftrightarrow \forall x_2 \, \forall x_3 \, R(f_1, x_2, x_3, f(x_2, x_3))
\end{array}$$

Thus for some f_1, f_4 (which we do not know much about)

$$(\mathcal{A}; f_1, f_4) \models \exists x_1 \forall x_2 \, \forall x_3 \, \exists x_4 \, R(x_1, x_2, x_3, x_4) \leftrightarrow \forall x_2 \, \forall x_3 \, R(f_1, x_2, x_3, f(x_2, x_3)). \; \Box$$

Remark 1.5 This lemma only states that for a given structure $\mathcal{A}$ and a given formula φ there is *some* expanded structure $\mathcal{B}$ in which the formula is equivalent to its Skolemization: $\mathcal{B} \models \varphi \leftrightarrow \varphi^S$. This equivalence need not hold in every expansion of $\mathcal{A}$:

Consider $\mathcal{A} = (\mathbb{N};)$ the structure with no relations (except $=$) and no functions. Expand it to $\mathcal{B}$ by adding the function $f(n) = n + 1$ as an interpretation of $F^{\mathcal{B}}$ Then

$$\mathcal{B} \models \forall x \exists y\, (x = y) \qquad \text{but} \qquad \mathcal{B} \not\models \forall x\, (x = F(x)).$$

Here $\forall x\, (x = F(x))$ is the Skolemization of $\forall x \exists y\, (x = y)$.

However, we do have

Lemma 1.6 *$\varphi^S \models \varphi$, for closed φ.* $\square$

Recall that $\varphi^S \models \varphi$ means that for any structure $\mathcal{B}$ in the language expanded with the Skolem symbols,

$$\mathcal{B} \models \varphi^S \Longrightarrow \mathcal{B} \models \varphi.$$

Hence a formula is implied by its Skolemization.

By the completeness theorem for predicate logic an L-formula φ is provable iff it is valid, i.e. in symbols

$$\vdash \varphi \qquad \Longleftrightarrow \qquad \models \varphi.$$

The right-hand side means that $\mathcal{A} \models \varphi$ for every L-structure $\mathcal{A}$.

The following result is the key to the resolution method:

Theorem 1.7 *Let ψ be any closed first-order formula. Suppose that φ is the prenex normal form of $\neg\psi$ and let φ^S be its Skolemization. Then*

$$\vdash \psi \qquad \Longleftrightarrow \qquad \vdash \neg(\varphi^S).$$

Proof. By the completeness theorem we may consider validity $\models$ instead of provability $\vdash$.

($\Rightarrow$) Suppose $\models \psi$. Since $\neg\psi$ is logically equivalent to φ, we have $\models \neg\varphi$. Suppose that

$$\not\models \neg\varphi^S$$

This means that there is some model $\mathcal{B}$ such that $\mathcal{B} \not\models \neg\varphi^S$, i.e. $\mathcal{B} \models \varphi^S$. But by Lemma 1.6, $\mathcal{B} \models \varphi$. This contradicts $\models \neg\varphi$. Hence

$$\models \neg\varphi^S$$

($\Leftarrow$) Suppose now $\models \neg\varphi^S$. For a contradiction assume $\not\models \psi$, i.e. that there exists some model $\mathcal{A}$ such that $\mathcal{A} \models \neg\psi$. Since $\neg\psi$ is logically equivalent to φ, we get $\mathcal{A} \models \varphi$. Hence by Lemma 1.4 there is some expansion of $\mathcal{A}$ such that

$$\mathcal{B} \models \varphi \leftrightarrow \varphi^S.$$

Now $\mathcal{B} \models \varphi$, since $\mathcal{B}$ is the same as $\mathcal{A}$ with respect to the symbols occuring in φ. Thereby $\mathcal{B} \models \varphi^S$ which contradicts the first assumption. We conclude that $\models \psi$. $\Box$

φ^S is equivalent to a conjunction of clauses $\gamma_1 \wedge \cdots \wedge \gamma_n$. A proof of $\neg\varphi^S$ can therefore considered as a derivation of a contradiction, i.e. an empty clause, from the set of clauses $\{\gamma_1, \ldots, \gamma_n\}$. A remarkable fact about the resolution method is that one derivation rule, albeit somewhat complicated, suffices.

1.2 Resolution

- From $A \vee L$ and $B \vee \neg L$, we may infer that $A \vee B$.

This is easily seen by considering the possible cases: $A \wedge B$, $A \wedge \neg L$, $L \wedge B$, and $L \wedge \neg L$. The first three cases each imply $A \vee B$ by $\vee$-introduction The last case is a contradiction, so $A \vee B$ follows then by the absurdity rule.

This rule may generalised as follows:

- Suppose that $\forall \bar{x}\,(A \vee L)$ and $\forall \bar{y}\,(B \vee \neg M)$ and $L^\theta = M^\theta$ for some subsitution θ of terms for variables. Then we may infer that $A^\theta \vee B^\theta$.

As instances of the assumptions, we have $A^\theta \vee L^\theta$ and, since $L^\theta = M^\theta$, also $B^\theta \vee \neg L^\theta$. We may now apply the former rule and get $A^\theta \vee B^\theta$.

If we let θ be the most general unifier (mgu) of L and M, then any other instance of the rule may be gotten out of this one. This is the idea of the resolution rule. We shall now work with sets of clauses and leave universal quantifiers implicit as in the previous section. A clause $\gamma = L_1 \vee \cdots \vee L_m$ is considered as a set of literals $\{L_1, \ldots, L_n\}$.

Definition 1.8 Let C and D be clauses with no variables in common. Let L and M be literals occuring in C and D respectively, and suppose that θ is an mgu for their atoms, and that L^θ and M^θ are complementary. Then the clause

$$(C^\theta - \{L^\theta\}) \cup (D^\theta - \{M^\theta\})$$

is called the *binary resolvent* of C and D. Its *parent clauses* are C and D.

Example 1.9 Consider the clauses $C = P(x) \vee \neg Q(g(x))$ and $D = Q(y) \vee R(y)$. Let $L = \neg Q(g(x))$ and $M = Q(y)$. The mgu of their atoms, i.e. of $Q(g(x))$ and $Q(y)$, is $\theta = \{y := g(x)\}$. The binary resolvent is

$$(C^\theta - \{L^\theta\}) \cup (D^\theta - \{M^\theta\}) = P(x)^\theta \vee R(y)\theta = P(x) \vee R(g(x)).$$

Example 1.10 The binary resolvent of $\neg Q(g(x))$ and $Q(y)$ is the empty clause $\Box$.

Example 1.11 To resolve $C = P(x) \vee \underline{\neg S(x, g(x))}$ and $D = \underline{S(h(y), z)} \vee R(z)$ we find an mgu of the atoms of the underlined literals, $\theta = \{x := h(y), z := g(h(y))\}$. The binary resolvent is then

$$P(x)^\theta \vee R(z)^\theta = P(h(y)) \vee R(g(h(y))).$$

Consider a pair of clauses $C = P(x, y) \vee Q(f(a)) \vee Q(z)$ and $D = \neg Q(f(y)) \vee \neg R(y)$. We may not do away with all the Q in one binary resolution step, although we see that $Q(f(a))$ and $Q(z)$ unify with $\theta = \{z := f(a)\}$. Considering C^θ as a factor of C, C^θ may now be binary resolved with D.

If a subset of the literals of a clause C has the same sign and mgu θ, then C^θ *is a factor of* C. Trivially C is a factor of itself

Definition 1.12 A *resolvent of clauses* C and D is a binary resolvent of C_1 and D_1 where C_1 is a factor of C and D_1 is a factor of D.

Thus $P(x, f(a)) \vee \neg R(a)$ is a resolvent of C and D in the example above.

Definition 1.13 Let S be a set of clauses. The *one-step resolution of* S is

$$\mathrm{Res}(S) = S \cup \{C : C \text{ is the resolvent of two clauses from } S\}$$

Resolution in several steps is defined recursively: Let

$$\begin{array}{rcl} \mathrm{Res}^0(S) & = & S \\ \mathrm{Res}^{n+1}(S) & = & \mathrm{Res}(\mathrm{Res}^n(S)). \end{array}$$

We have

$$S \subseteq \mathrm{Res}(S) \subseteq \mathrm{Res}^2(S) \subseteq \mathrm{Res}^3(S) \subseteq \cdots \subseteq \mathrm{Res}^n(S) \subseteq \cdots$$

This sequence may in general grow for ever as in the following example.

Example 1.14 Let $S = \{\neg P(x) \vee P(f(x)), P(0)\}$. Then

$$\begin{array}{rcl} S & = & \{\underline{\neg P(x)} \vee P(f(x)), \underline{P(0)}\} \\ \mathrm{Res}(S) & = & \{\underline{\neg P(x)} \vee P(f(x)), P(0), \underline{P(f(0))}\} \\ \mathrm{Res}^2(S) & = & \{\underline{\neg P(x)} \vee P(f(x)), P(0), P(f(0)), \underline{P(f(f(0)))}\} \\ & \vdots & \\ \mathrm{Res}^n(S) & = & \{\underline{\neg P(x)} \vee P(f(x)), P(0), P(f(0))), P(f(f(0))), \ldots, \underline{P(f^n(0))}\} \\ & \vdots & \end{array}$$

Theorem 1.15 (The Resolution Theorem, Robinson 1965) *Let $M = \{C_1, \dots, C_m\}$ be a set of clauses with free variables among $\bar{x}$. Then*

$$\vdash \neg\forall\bar{x} \bigwedge_{i=1}^{m} C_i \qquad \textit{if and only if} \qquad \Box \in \mathrm{Res}^n(M) \textit{ for some } n \geq 0$$

To sum up we get the following method for proving a closed first-order formula ψ. Let φ be the PNF of $\neg\psi$ and let φ^S be the Skolemization of φ. Using the procedure for finding CNF we may rewrite

$$\varphi^S \leftrightarrow \forall\bar{x} \bigwedge_{i=1}^{m} C_i$$

where each C_i is a clause. By Theorem 1.7 and the Resolution Theorem we have the following equivalences for $M = \{C_1, \dots, C_n\}$

$$\begin{aligned} \vdash \psi &\iff \vdash \neg\varphi^S \\ &\iff \vdash \neg\forall\bar{x} \bigwedge_{i=1}^{m} C_i \\ &\iff \Box \in \mathrm{Res}^n(M) \text{ for some } n \geq 0 \end{aligned}$$

References

S.K. Das. *Deductive Databases and Logic Programming.* Addison-Wesley 1992.

K.B. Hansen. *Grundläggande logik.* 3:e uppl. Studentlitteratur 1997.