Proof. If reduction rules are preferred over inference rules, then the overall length if a clause cannot exceed n, where n is the number of variables. Multiple occurrences of the same literal are removed by rule Condensation, multiple occurrences of the same variable with different sign result in an application of rule Tautology Deletion. The number of such clauses can be overestimated by 3^n because every variable occurs at most once positively, negatively or not at all in clause. Hence, there are at most $2n3^n$ different resolution applications. $\square$

Of course, what needs to be shown is that the strategy employed in Theorem 2.6.5 is still complete. This is not completely trivial and gets very nasty using semantic trees as the proof method of choice. So I postpone this proof until superposition is established where this result becomes a particular instance of superposition completeness. Exercise **??** contains the completeness part when the reduction rules are preferred over inference rules.

2.7 Propositional Superposition

Superposition was originally developed for first-order logic with equality [1]. Here I introduce its projection to propositional logic. Compared to the resolution calculus superposition adds (i) ordering and selection restrictions on inferences, (ii) an abstract redundancy notion, (iii) the notion of a partial model, based on the ordering for inference guidance, and (iv) a *saturation* concept.

Definition 2.7.1 (Clause Ordering)**.** Let $\prec$ be a total strict ordering on Σ. Then $\prec$ can be lifted to a total ordering on literals by $\prec \subseteq \prec_L$ and $P \prec_L \neg P$ and $\neg P \prec_L Q$, $\neg P \prec_L \neg Q$ for all $P \prec Q$. The ordering $\prec_L$ can be lifted to a total ordering on clauses $\prec_C$ by considering the multiset extension of $\prec_L$ for clauses.

For example, if $P \prec Q$, then $P \prec_L \neg P \prec_L Q \prec_L \neg Q$ and $P \vee Q \prec_C P \vee Q \vee Q \prec_C \neg Q$ because $\{P, Q\} \prec_L^{\mathrm{mul}} \{P, Q, Q\} \prec_L^{\mathrm{mul}} \{\neg Q\}$.

Proposition 2.7.2 (Properties of the Clause Ordering)**.** (i) The orderings on literals and clauses are total and well-founded.
(ii) Let C and D be clauses with $P = \mathrm{atom}(\max(C))$, $Q = \mathrm{atom}(\max(D))$, where $\max(C)$ denotes the maximal literal in C.

1. If $Q \prec_L P$ then $D \prec_C C$.

2. If $P = Q$, P occurs negatively in C but only positively in D, then $D \prec_C C$.

Eventually, I overload $\prec$ with $\prec_L$ and $\prec_C$. So if $\prec$ is applied to literals it denotes $\prec_L$, if it is applied to clauses, it denotes $\prec_C$. Note that $\prec$ is a total ordering on literals and clauses as well. Eventually we will restrict inferences to maximal literals with respect to $\prec$. For a clause set N, I define $N^{\prec C} = \{D \in N \mid D \prec C\}$.

Definition 2.7.3 (Abstract Redundancy)**.** A clause C is *redundant* with respect to a clause set N if $N^{\prec C} \models C$.

Tautologies are redundant. Subsumed clauses are redundant if $\subseteq$ is strict. Duplicate clauses are anyway eliminated quietly because the calculus operates on sets of clauses.

Note that for finite N, and any $C \in N$ redundancy $N^{\prec C} \models C$ can be decided but is as hard as testing unsatisfiability for a clause set N. So the goal is to invent redundancy notions that can be efficiently decided and that are useful.

Definition 2.7.4 (Selection Function)**.** The selection function sel maps clauses to one of its negative literals or $\bot$. If $\operatorname{sel}(C) = \neg P$ then $\neg P$ is called *selected* in C. If $\operatorname{sel}(C) = \bot$ then no literal in C is *selected*.

The selection function is, in addition to the ordering, a further means to restrict superposition inferences. If a negative literal is selected on a clause, any superposition inference must be on the selected literal.

Definition 2.7.5 (Partial Model Construction)**.** Given a clause set N and an ordering $\prec$ we can construct a (partial) Herbrand model $N_{\mathcal{I}}$ for N inductively as follows:

$$
\begin{array}{rcl}
N_C & := & \bigcup_{D \prec C} \delta_D \\[1ex]
\delta_D & := & \begin{cases} \{P\} & \text{if } D = D' \vee P, P \text{ strictly maximal, no literal} \\ & \text{selected in } D \text{ and } N_D \not\models D \\ \emptyset & \text{otherwise} \end{cases} \\
N_{\mathcal{I}} & := & \bigcup_{C \in N} \delta_C
\end{array}
$$

Clauses C with $\delta_C \neq \emptyset$ are called *productive*.

Proposition 2.7.6. Some properties of the partial model construction.

1. For every D with $(C \vee \neg P) \prec D$ we have $\delta_D \neq \{P\}$.

2. If $\delta_C = \{P\}$ then $N_C \cup \delta_C \models C$.

3. If $N_C \models D$ and $D \prec C$ then for all C' with $C \prec C'$ we have $N_{C'} \models D$ and in particular $N_{\mathcal{I}} \models D$.

4. There is no clause C with $P \vee P \prec C$ such that $\delta_C = \{P\}$.

T

Please properly distinguish: N is a set of clauses interpreted as the conjunction of all clauses. $N^{\prec C}$ is of set of clauses from N strictly smaller than C with respect to $\prec$. $N_{\mathcal{I}}$, N_C are sets of atoms, often called *Herbrand Interpretations*. $N_{\mathcal{I}}$ is the overall (partial) model for N, whereas N_C is generated from all clauses from N strictly smaller than C. Validity is defined by $N_{\mathcal{I}} \models P$ if $P \in N_{\mathcal{I}}$ and $N_{\mathcal{I}} \models \neg P$ if $P \notin N_{\mathcal{I}}$, accordingly for N_C.

Given some clause set N the partial model $N_{\mathcal{I}}$ can be extended to a valuation $\mathcal{A}$ by defining $\mathcal{A}(N_{\mathcal{I}}) := N_{\mathcal{I}} \cup \{\neg P \mid P \notin N_{\mathcal{I}}\}$. So we can also define for some Herbrand interpretation $N_{\mathcal{I}}$ (N_C) that $N_{\mathcal{I}} \models \phi$ iff $\mathcal{A}(N_{\mathcal{I}})(\phi) = 1$.

Superposition Left $\quad (N \uplus \{C_1 \vee P, C_2 \vee \neg P\}) \Rightarrow_{\mathrm{SUP}} (N \cup \{C_1 \vee P, C_2 \vee \neg P\} \cup \{C_1 \vee C_2\})$

where (i) P is strictly maximal in $C_1 \vee P$ (ii) no literal in $C_1 \vee P$ is selected (iii) $\neg P$ is maximal and no literal selected in $C_2 \vee \neg P$, or $\neg P$ is selected in $C_2 \vee \neg P$

Factoring $\quad (N \uplus \{C \vee P \vee P\}) \Rightarrow_{\mathrm{SUP}} (N \cup \{C \vee P \vee P\} \cup \{C \vee P\})$

where (i) P is maximal in $C \vee P \vee P$ (ii) no literal is selected in $C \vee P \vee P$

Note that the superposition factoring rule differs from the resolution factoring rule in that it only applies to positive literals. Abstract redundancy can also be lifted to inferences, in the propositional case to Superposition Left applications. A Superposition Left inference

$$(N \uplus \{C_1 \vee P, C_2 \vee \neg P\}) \Rightarrow_{\mathrm{SUP}} (N \cup \{C_1 \vee P, C_2 \vee \neg P\} \cup \{C_1 \vee C_2\})$$

is redundant if either one of the clauses $C_1 \vee P, C_2 \vee \neg P$ is redundant, or if $N^{\prec C_2 \vee \neg P} \models C_1 \vee C_2$. For a Factoring inference, the conclusion $C \vee P$ makes the premise $C \vee P \vee P$, so it is sufficient to require that $C \vee P \vee P$ is not redundant in order to guarantee $C \vee P$ to be non-redundant.

Definition 2.7.7 (Saturation)**.** A set N of clauses is called *saturated up to redundancy*, if any inference from non-redundant clauses in N yields a redundant clause with respect to N or is already contained in N.

Alternatively, saturation can be defined on the basis of redundant inferences: a set N is saturated up to redundancy if all inferences from clauses from N are redundant. Examples for specific redundancy rules that can be efficiently decided are

Subsumption $\quad (N \uplus \{C_1, C_2\}) \Rightarrow_{\mathrm{SUP}} (N \cup \{C_1\})$

provided $C_1 \subset C_2$

Tautology Deletion $\quad (N \uplus \{C \vee P \vee \neg P\}) \Rightarrow_{\mathrm{SUP}} (N)$

Condensation $\quad (N \uplus \{C_1 \vee L \vee L\}) \Rightarrow_{\mathrm{SUP}} (N \cup \{C_1 \vee L\})$