and if $S(D\sigma) \simeq S(D)$, $S(C\rho) \simeq S(C)$ (that is, “corresponding” literals are selected), then there exists a substitution $\tau$ such that

\[
\frac{D}{C'} \quad \frac{C''}{\tau} \\
\downarrow \tau
\]

[inference in $\text{Res}_S^>$]

\[C' = C''\tau\]

An analogous lifting lemma holds for factorization.

**Saturation of General Clause Sets**

**Corollary 3.42** Let $N$ be a set of general clauses saturated under $\text{Res}_S^>$, i.e., $\text{Res}_S^>(N) \subseteq N$. Then there exists a selection function $S'$ such that $S|_N = S'|_N$ and $G_S(N)$ is also saturated, i.e.,

\[\text{Res}_S^>(G_S(N)) \subseteq G_S(N).\]

**Proof.** We first define the selection function $S'$ such that $S'(C) = S(C)$ for all clauses $C \in G_S(N) \cap N$. For $C \in G_S(N) \setminus N$ we choose a fixed but arbitrary clause $D \in N$ with $C \in G_S(D)$ and define $S'(C)$ to be those occurrences of literals that are ground instances of the occurrences selected by $S$ in $D$. Then proceed as in the proof of Corollary 3.34 using the above lifting lemma.

**Soundness and Refutational Completeness**

**Theorem 3.43** Let $\succ$ be an atom ordering and $S$ a selection function such that $\text{Res}_S^>(N) \subseteq N$. Then

\[N \models \bot \iff \bot \in N\]

**Proof.** The “$\Leftarrow$” part is trivial. For the “$\Rightarrow$” part consider first the propositional level: Construct a candidate interpretation $I_N$ as for unrestricted resolution, except that clauses $C$ in $N$ that have selected literals are not productive, even when they are false in $I_C$ and when their maximal atom occurs only once and positively. The result for general clauses follows using Corollary 3.42.

72
Craig-Interpolation

A theoretical application of ordered resolution is Craig-Interpolation:

**Theorem 3.44 (Craig 1957)** Let $F$ and $G$ be two propositional formulas such that $F \models G$. Then there exists a formula $H$ (called the interpolant for $F \models G$), such that $H$ contains only prop. variables occurring both in $F$ and in $G$, and such that $F \models H$ and $H \models G$.

**Proof.** Translate $F$ and $\neg G$ into CNF. let $N$ and $M$, resp., denote the resulting clause set. Choose an atom ordering $\succ$ for which the prop. variables that occur in $F$ but not in $G$ are maximal. Saturate $N$ into $N^*$ w. r. t. $Res^\succ_S$ with an empty selection function $S$. Then saturate $N^* \cup M$ w. r. t. $Res^\succ_S$ to derive $\bot$. As $N^*$ is already saturated, due to the ordering restrictions only inferences need to be considered where premises, if they are from $N^*$, only contain symbols that also occur in $G$. The conjunction of these premises is an interpolant $H$. The theorem also holds for first-order formulas. For universal formulas the above proof can be easily extended. In the general case, a proof based on resolution technology is more complicated because of Skolemization. □

Redundancy

So far: local restrictions of the resolution inference rules using orderings and selection functions.

Is it also possible to delete clauses altogether? Under which circumstances are clauses unnecessary? (Conjecture: e. g., if they are tautologies or if they are subsumed by other clauses.)

Intuition: If a clause is guaranteed to be neither a minimal counterexample nor productive, then we do not need it.

A Formal Notion of Redundancy

Let $N$ be a set of ground clauses and $C$ a ground clause (not necessarily in $N$). $C$ is called redundant w. r. t. $N$, if there exist $C_1, \ldots, C_n \in N$, $n \geq 0$, such that $C_i \prec C$ and $C_1, \ldots, C_n \models C$.

Redundancy for general clauses: $C$ is called redundant w. r. t. $N$, if all ground instances $C\sigma$ of $C$ are redundant w. r. t. $G_\Sigma(N)$.

Intuition: Redundant clauses are neither minimal counterexamples nor productive.

Note: The same ordering $\prec$ is used for ordering restrictions and for redundancy (and for the completeness proof).
Examples of Redundancy

Proposition 3.45 Some redundancy criteria:

- $C$ tautology (i.e., $| C | \Rightarrow C$ redundant w.r.t. any set $N$.
- $C \sigma \subset D \Rightarrow D$ redundant w.r.t. $N \cup \{ C \}$.
- $C \sigma \subseteq D \Rightarrow D \lor \neg \sigma$ redundant w.r.t. $N \cup \{ C \lor L, D \}$.

(Under certain conditions one may also use non-strict subsumption, but this requires a slightly more complicated definition of redundancy.)

Saturation up to Redundancy

$N$ is called saturated up to redundancy (w.r.t. $Res \succeq S$):

$$\Leftrightarrow Res \succeq S (N \setminus Red(N)) \subseteq N \cup Red(N)$$

Theorem 3.46 Let $N$ be saturated up to redundancy. Then

$N \models \bot \Leftrightarrow \bot \in N$

Proof (Sketch). (i) Ground case:

- consider the construction of the candidate interpretation $I_N \succeq N$ for $Res \succeq S$
- redundant clauses are not productive
- redundant clauses in $N$ are not minimal counterexamples for $I_N \succeq N$

The premises of “essential” inferences are either minimal counterexamples or productive.

(ii) Lifting: no additional problems over the proof of Theorem 3.43.

Monotonicity Properties of Redundancy

Theorem 3.47

(i) $N \subseteq M \Rightarrow Red(N) \subseteq Red(M)$
(ii) $M \subseteq Red(N) \Rightarrow Red(N) \subseteq Red(N \setminus M)$

Proof. Exercise.
A Resolution Prover

So far: static view on completeness of resolution:

Saturated sets are inconsistent if and only if they contain $\bot$.

We will now consider a dynamic view:

How can we get saturated sets in practice?

The theorems 3.46 and 3.47 are the basis for the completeness proof of our prover $RP$.

Rules for Simplifications and Deletion

We want to employ the following rules for simplification of prover states $N$:

- *Deletion of tautologies*

  $$N \cup \{C \lor A \lor \neg A\} \triangleright N$$

- *Deletion of subsumed clauses*

  $$N \cup \{C, D\} \triangleright N \cup \{C\}$$

  if $C\sigma \subseteq D$ ($C$ subsumes $D$).

- *Reduction* (also called subsumption resolution)

  $$N \cup \{C \lor L, D \lor C\sigma \lor L\sigma\} \triangleright N \cup \{C \lor L, D \lor C\sigma\}$$

Resolution Prover $RP$

3 clause sets: $N$(ew) containing new resolvents

$P$(rocessed) containing simplified resolvents

clauses get into $O$(ld) once their inferences have been computed

Strategy: Inferences will only be computed when there are no possibilities for simplification
Transition Rules for $RP$ (I)

Tautology elimination
\[ N \cup \{C\} \mid P \mid O \quad \Rightarrow \quad N \mid P \mid O \]
if $C$ is a tautology

Forward subsumption
\[ N \cup \{C\} \mid P \mid O \quad \Rightarrow \quad N \mid P \mid O \]
if some $D \in P \cup O$ subsumes $C$

Backward subsumption
\[ N \cup \{C\} \mid P \setminus \{D\} \mid O \quad \Rightarrow \quad N \cup \{C\} \mid P \mid O \]
\[ N \cup \{C\} \mid P \setminus \{C\} \mid O \quad \Rightarrow \quad N \cup \{C\} \mid P \mid O \]
if $C$ strictly subsumes $D$

Transition Rules for $RP$ (II)

Forward reduction
\[ N \cup \{C \lor L\} \mid P \mid O \quad \Rightarrow \quad N \cup \{C\} \mid P \mid O \]
if there exists $D \lor L' \in P \cup O$

such that $\overline{L} = L'\sigma$ and $D\sigma \subseteq C$

Backward reduction
\[ N \mid P \cup \{C \lor L\} \mid O \quad \Rightarrow \quad N \mid P \cup \{C\} \mid O \]
\[ N \mid P \setminus \{C\} \mid O \quad \Rightarrow \quad N \mid P \cup \{C\} \mid O \]
if there exists $D \lor L' \in N$

such that $\overline{L} = L'\sigma$ and $D\sigma \subseteq C$

Transition Rules for $RP$ (III)

Clause processing
\[ N \cup \{C\} \mid P \mid O \quad \Rightarrow \quad N \mid P \cup \{C\} \mid O \]

Inference computation
\[ \emptyset \mid P \cup \{C\} \mid O \quad \Rightarrow \quad N \mid P \mid O \cup \{C\}, \]
\[ \text{with } N = Res_S(O \cup \{C\}) \]

Soundness and Completeness

Theorem 3.48
\[ N \models \bot \iff N \mid \emptyset \mid \emptyset \quad \Rightarrow \quad N' \cup \{\bot\} \mid \_ \mid \_ \]

**Fairness**

Problem:

If $N$ is inconsistent, then $N \vdash \emptyset \vdash^* N' \cup \{\bot\} \vdash \bot$.

Does this imply that every derivation starting from an inconsistent set $N$ eventually produces $\bot$?

No: a clause could be kept in $P$ without ever being used for an inference.

We need in addition a *fairness condition*:

If an inference is possible forever (that is, none of its premises is ever deleted), then it must be computed eventually.

One possible way to guarantee fairness: Implement $P$ as a queue (there are other techniques to guarantee fairness).

With this additional requirement, we get a stronger result: If $N$ is inconsistent, then every fair derivation will eventually produce $\bot$.

**Hyperresolution**

There are many variants of resolution. (We refer to [Bachmair, Ganzinger: Resolution Theorem Proving] for further reading.)

One well-known example is hyperresolution (Robinson 1965):

Assume that several negative literals are selected in a clause $C$. If we perform an inference with $C$, then one of the selected literals is eliminated.

Suppose that the remaining selected literals of $C$ are again selected in the conclusion. Then we must eliminate the remaining selected literals one by one by further resolution steps.

Hyperresolution replaces these successive steps by a single inference. As for $Res^+\subseteq S$, the calculus is parameterized by an atom ordering $\succ$ and a selection function $S$.

\[
\frac{D_1 \lor B_1 \ldots \lor D_n \lor B_n \quad C \lor \neg A_1 \lor \ldots \lor \neg A_n}{(D_1 \lor \ldots \lor D_n \lor C)\sigma}
\]

with $\sigma = \text{mgu}(A_1 \doteq B_1, \ldots, A_n \doteq B_n)$, if

(i) $B_i\sigma$ strictly maximal in $D_i\sigma$, $1 \leq i \leq n$;

(ii) nothing is selected in $D_i$;