3.8 Inference Systems and Proofs

Inference systems Γ (proof calculi) are sets of tuples

$$(F_1, \dots, F_n, F_{n+1}),\ n \geq 0,$$

called *inferences*, and written

$$\frac{\overbrace{F_1 \ \dots \ F_n}^{\text{premises}}}{\underbrace{F_{n+1}}_{\text{conclusion}}} \quad \textit{side condition}$$

Clausal inference system: premises and conclusions are clauses. One also considers inference systems over other data structures.

Inference Systems

Inference systems Γ are shorthands for reduction systems over sets of formulas. If N is a set of formulas, then

$$\frac{\overbrace{F_1 \ \dots \ F_n}^{\text{premises}}}{\underbrace{F_{n+1}}_{\text{conclusion}}} \quad \textit{side condition}$$

is a shorthand for

$$N \cup \{F_1, \dots, F_n\} \quad \Rightarrow_\Gamma \quad N \cup \{F_1, \dots, F_n\} \cup \{F_{n+1}\}$$
$$\text{if } \textit{side condition}$$

Proofs

A *proof* in Γ of a formula F from a set of formulas N (called *assumptions*) is a sequence $F_1, \dots, F_k$ of formulas where

(i) $F_k = F$,

(ii) for all $1 \leq i \leq k$: $F_i \in N$ or there exists an inference

$$\frac{F_{m_1} \ \dots \ F_{m_n}}{F_i}$$

in Γ, such that $0 \leq m_j < i$, for $1 \leq j \leq n$.

Soundness and Completeness

Provability $\vdash_\Gamma$ of F from N in Γ:

$N \vdash_\Gamma F$ if there exists a proof in Γ of F from N.

Γ is called *sound*, if

$$\frac{F_1 \ \ldots \ F_n}{F} \in \Gamma \ \text{ implies } F_1, \ldots, F_n \models F$$

Γ is called *complete*, if

$N \models F$ implies $N \vdash_\Gamma F$

Γ is called *refutationally complete*, if

$N \models \bot$ implies $N \vdash_\Gamma \bot$

Proposition 3.13

(i) Let Γ be sound. Then $N \vdash_\Gamma F \Rightarrow N \models F$

(ii) If $N \vdash_\Gamma F$ then there exist finitely many $F_1, \ldots, F_n \in N$ such that $F_1, \ldots, F_n \vdash_\Gamma F$

Reduced Proofs

The definition of a proof of F given above admits sequences $F_1, \ldots, F_k$ of formulas where some F_i are not ancestors of $F_k = F$ (i. e., some F_i are not actually used to derive F).

A proof is called *reduced*, if every F_i with $i < k$ is an ancestor of F_k.

We obtain a reduced proof from a proof by marking first F_k and then recursively all the premises used to derive a marked conclusion, and by deleting all non-marked formulas in the end.

Reduced Proofs as Trees

$$
\begin{array}{rcl}
\text{markings} & \widehat{=} & \text{formulas} \\
\text{leaves} & \widehat{=} & \text{assumptions and axioms} \\
\text{other nodes} & \widehat{=} & \text{inferences:} \quad \begin{array}{rcl} \text{conclusion} & \widehat{=} & \text{parent node} \\ \text{premises} & \widehat{=} & \text{child nodes} \end{array}
\end{array}
$$

$$
\dfrac{P(f(c)) \qquad \dfrac{\dfrac{\dfrac{P(f(c)) \lor Q(b) \qquad \dfrac{\dfrac{P(f(c)) \lor Q(b) \quad \neg P(f(c)) \lor \neg P(f(c)) \lor Q(b)}{\neg P(f(c)) \lor Q(b) \lor Q(b)}}{\neg P(f(c)) \lor Q(b)}}{Q(b) \lor Q(b)}}{Q(b)} \qquad \neg P(f(c)) \lor \neg Q(b)}{\neg P(f(c))}}{\perp}
$$

Mandatory vs. Admissible Inferences

It is useful to distinguish between two kinds of inferences:

- Mandatory (required) inferences:

  Must be performed to ensure refutational completeness.

  The less, the better.

- Optional (admissible) inferences:

  May be performed, if useful.

We will first consider only mandatory inferences.

3.9 Ground (or propositional) Resolution

We observe that propositional clauses and ground clauses are essentially the same, as long as we do not consider equational atoms.

In this section we only deal with ground clauses.

Unlike in Section 2 we admit duplicated literals in clauses, i. e., we treat clauses like multisets of literals, not like sets.

The Resolution Calculus *Res*

Resolution inference rule:

$$\frac{D \vee A \qquad C \vee \neg A}{D \vee C}$$

Terminology: $D \vee C$: *resolvent*; A: *resolved atom*

(Positive) factorization inference rule:

$$\frac{C \vee A \vee A}{C \vee A}$$

These are *schematic inference rules*; for each substitution of the *schematic variables* C, D, and A, by ground clauses and ground atoms, respectively, we obtain an inference.

We treat "$\vee$" as associative and commutative, hence A and $\neg A$ can occur anywhere in the clauses; moreover, when we write $C \vee A$, etc., this includes unit clauses, that is, $C = \bot$.

Sample Refutation

1	$\neg P(f(c)) \vee \neg P(f(c)) \vee Q(b)$	(given)
2	$P(f(c)) \vee Q(b)$	(given)
3	$\neg P(g(b,c)) \vee \neg Q(b)$	(given)
4	$P(g(b,c))$	(given)
5	$\neg P(f(c)) \vee Q(b) \vee Q(b)$	(Res. 2 into 1)
6	$\neg P(f(c)) \vee Q(b)$	(Fact. 5)
7	$Q(b) \vee Q(b)$	(Res. 2 into 6)
8	$Q(b)$	(Fact. 7)
9	$\neg P(g(b,c))$	(Res. 8 into 3)
10	$\bot$	(Res. 4 into 9)

Soundness of Resolution

Theorem 3.14 *Ground first-order resolution is sound.*

Proof. As in propositional logic. □

Note: In ground first-order logic we have (like in propositional logic):

1. $\mathcal{B} \models L_1 \vee \ldots \vee L_n$ if and only if there exists i: $\mathcal{B} \models L_i$.
2. $\mathcal{B} \models A$ or $\mathcal{B} \models \neg A$.

This does not hold for formulas with variables!

3.10 Refutational Completeness of Resolution

How to show refutational completeness of ground resolution:

- We have to show: $N \models \bot \;\Rightarrow\; N \vdash_{Res} \bot$, or equivalently: If $N \not\vdash_{Res} \bot$, then N has a model.
- Idea: Suppose that we have computed sufficiently many inferences (and not derived $\bot$).
- Now order the clauses in N according to some appropriate ordering, inspect the clauses in ascending order, and construct a series of Herbrand interpretations.
- The limit interpretation can be shown to be a model of N.

Closure of Clause Sets under *Res*

$$
\begin{aligned}
Res(N) &= \{\, C \mid C \text{ is conclusion of an inference in } Res \\
&\qquad\qquad \text{with premises in } N \,\} \\
Res^0(N) &= N \\
Res^{n+1}(N) &= Res(Res^n(N)) \cup Res^n(N), \text{ for } n \geq 0 \\
Res^*(N) &= \textstyle\bigcup_{n \geq 0} Res^n(N)
\end{aligned}
$$

N is called *saturated* (w. r. t. resolution), if $Res(N) \subseteq N$.

Proposition 3.15

(i) $Res^*(N)$ is saturated.

(ii) Res is refutationally complete, if and only if for each set N of ground clauses:

$$N \models \bot \;\text{ implies }\; \bot \in Res^*(N)$$

Proof. (i): We have to show that $Res(Res^*(N)) \subseteq Res^*(N)$, or in other words, that the conclusion of every inference in Res with premises in $Res^*(N)$ is again contained in $Res^*(N)$. An inference in Res is either a resolution inference or a factorization inference. Let us first consider a resolution inference with premises $C_1 \in Res^*(N)$ and $C_2 \in Res^*(N)$ and conclusion C. Since $Res^*(N) = \bigcup_{n\geq 0} Res^n(N)$, we know that there exist $j, k \geq 0$ such that $C_1 \in Res^j(N)$ and $C_2 \in Res^k(N)$. Without loss of generality assume that $j \geq k$. It is easy to see that in this case $Res^k(N) \subseteq Res^j(N)$, hence $C_1 \in Res^j(N)$ and $C_2 \in Res^j(N)$. Consequently, $C \in Res(Res^j(N)) \subseteq Res^{j+1}(N) \subseteq Res^*(N)$.

Otherwise we have a factorization inference with premise $C_1 \in Res^*(N)$ and conclusion C. Again we conclude that $C_1 \in Res^j(N)$ for some $j \geq 0$, hence $C \in Res(Res^j(N)) \subseteq Res^{j+1}(N) \subseteq Res^*(N)$.

(ii) This part follows immediately from the fact that for every clause C we have $N \vdash_{Res} C$ if and only if $C \in Res^*(N)$. $\square$

Clause Orderings

1. We assume that $\succ$ is any fixed ordering on ground atoms that is *total* and *well-founded*. (There exist many such orderings, e. g., the length-based ordering on atoms when these are viewed as words over a suitable alphabet.)

2. Extend $\succ$ to an *ordering* $\succ_L$ *on ground literals:*

$$\begin{array}{lcll} [\neg]A & \succ_L & [\neg]B & \text{, if } A \succ B \\ \neg A & \succ_L & A & \end{array}$$

3. Extend $\succ_L$ to an *ordering* $\succ_C$ *on ground clauses:*
$\succ_C = (\succ_L)_{\text{mul}}$, the multiset extension of $\succ_L$.

   Notation: $\succ$ also for $\succ_L$ and $\succ_C$.

Example

Suppose $A_5 \succ A_4 \succ A_3 \succ A_2 \succ A_1 \succ A_0$. Then:

$$\begin{array}{cc} & A_1 \vee \neg A_5 \\ \succ & A_3 \vee \neg A_4 \\ \succ & \neg A_1 \vee A_3 \vee A_4 \\ \succ & A_1 \vee \neg A_2 \\ \succ & \neg A_1 \vee A_2 \\ \succ & A_1 \vee A_1 \vee A_2 \\ \succ & A_0 \vee A_1 \end{array}$$

Properties of the Clause Ordering

Proposition 3.16

1. *The orderings on literals and clauses are total and well-founded.*

2. *Let C and D be clauses with $A = \text{maxatom}(C)$, $B = \text{maxatom}(D)$, where* $\text{maxatom}(C)$ *denotes the maximal atom in C.*

   (i) If $A \succ B$ then $C \succ D$.

   (ii) If $A = B$, A occurs negatively in C but only positively in D, then $C \succ D$.

Stratified Structure of Clause Sets

Let $A \succ B$. Clause sets are then stratified in this form:

$$
\succ \qquad
\begin{array}{r|l|l}
 & & \\ \hline
A \left\{ \begin{array}{c} \\ \\ \\ \end{array} \right. & \begin{array}{l} \neg A \vee \ldots \\ \ldots \vee A \vee A \\ \ldots \vee A \end{array} & \text{all clauses } C \text{ with maxatom}(C) = A \\ \hline
 & \quad\vdots & \\ \hline
B \left\{ \begin{array}{c} \\ \\ \\ \end{array} \right. & \begin{array}{l} \neg B \vee \ldots \\ \ldots \vee B \vee B \\ \ldots \vee B \end{array} & \text{all clauses } D \text{ with maxatom}(D) = B \\ \hline
 & & \\
\end{array}
$$

Construction of Interpretations

Given: set N of ground clauses, atom ordering $\succ$.

Wanted: Herbrand interpretation I such that

$$I \models N \quad \text{if } N \text{ is saturated and } \bot \notin N$$

Construction according to $\succ$, starting with the smallest clause.

Main Ideas of the Construction

- Clauses are considered in the order given by $\succ$.
- When considering C, one already has an interpretation so far available (I_C). Initially $I_C = \emptyset$.
- If C is true in this interpretation, nothing needs to to be changed.
- Otherwise, one would like to change the interpretation such that C becomes true.
- Changes should, however, be *monotone*. One never deletes atoms from the interpretation, and the truth value of clauses smaller than C should not change from true to false.
- Hence, one adds $\Delta_C = \{A\}$, if and only if C is false in I_C, if A occurs positively in C (*adding A will make C become true*) and if this occurrence in C is strictly maximal in the ordering on literals (*changing the truth value of A has no effect on smaller clauses*). Otherwise, $\Delta_C = \emptyset$.

- We say that the construction fails for a clause C, if C is false in I_C and $\Delta_C = \emptyset$.
- We will show: If there are clauses for which the construction fails, then some inference with the smallest such clause (the so-called "minimal counterexample") has not been computed. Otherwise, the limit interpretation is a model of all clauses.

Construction of Candidate Interpretations

Let $N, \succ$ be given. We define sets I_C and Δ_C for all ground clauses C over the given signature inductively over $\succ$:

$$I_C \quad := \quad \bigcup_{C \succ D} \Delta_D$$

$$\Delta_C \quad := \quad \begin{cases} \{A\}, & \text{if } C \in N,\ C = C' \vee A,\ A \succ C',\ I_C \not\models C \\ \emptyset, & \text{otherwise} \end{cases}$$

We say that C *produces* A, if $\Delta_C = \{A\}$.

Note that the definitions satisfy the conditions of Thm. 1.8; so they are well-defined even if $\{\, D \mid C \succ D \,\}$ is infinite.

The *candidate interpretation* for N (w. r. t. $\succ$) is given as $I_N^{\succ} := \bigcup_C \Delta_C$. (We also simply write I_N or I for $I_N^{\succ}$ if $\succ$ is either irrelevant or known from the context.)

Example

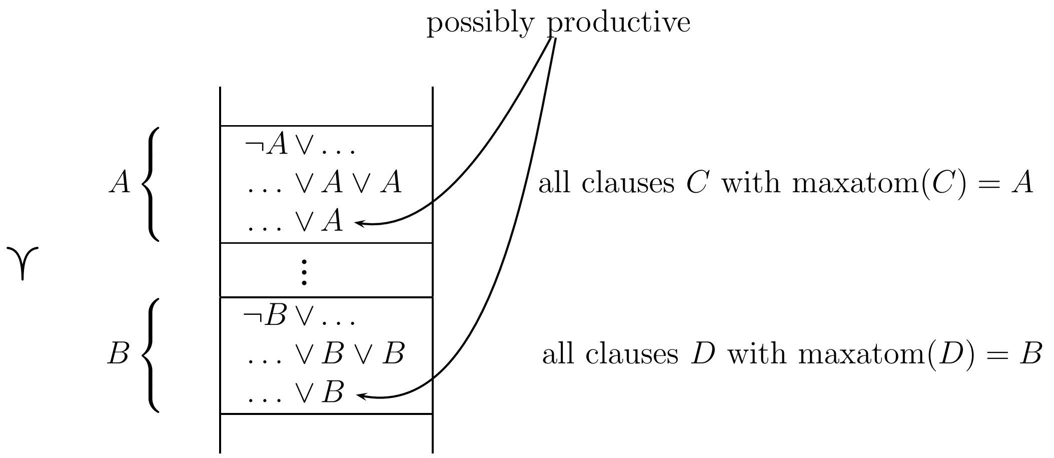
Let $A_5 \succ A_4 \succ A_3 \succ A_2 \succ A_1 \succ A_0$ (max. literals in red)

	clauses C	I_C	Δ_C	Remarks
7	$\neg A_1 \vee A_5$	$\{A_1, A_2, A_4\}$	$\{A_5\}$	
6	$\neg A_1 \vee A_3 \vee \neg A_4$	$\{A_1, A_2, A_4\}$	$\emptyset$	max. lit. $\neg A_4$ neg.; *min. counter-ex.*
5	$A_0 \vee \neg A_1 \vee A_3 \vee A_4$	$\{A_1, A_2\}$	$\{A_4\}$	A_4 maximal
4	$\neg A_1 \vee A_2$	$\{A_1\}$	$\{A_2\}$	A_2 maximal
3	$A_1 \vee A_2$	$\{A_1\}$	$\emptyset$	true in I_C
2	$A_0 \vee A_1$	$\emptyset$	$\{A_1\}$	A_1 maximal
1	$\neg A_0$	$\emptyset$	$\emptyset$	true in I_C

$I = \{A_1, A_2, A_4, A_5\}$ is not a model of the clause set
$\Rightarrow$ there exists a *counterexample*.

Structure of $N, \succ$

Let $A \succ B$. Note that producing a new atom does not change the truth value of smaller clauses.



Some Properties of the Construction

Proposition 3.17

(i) *If $D = D' \vee \neg A$, then no $C \succeq D$ produces A.*

(ii) *If $I_D \models D$, then $I_C \models D$ for every $C \succeq D$ and $I_N^{\succ} \models D$.*

(iii) *If $D = D' \vee A$ produces A, then $I_C \models D$ for every $C \succ D$ and $I_N^{\succ} \models D$.*

(iv) *If $D = D' \vee A$ produces A, then $I_C \not\models D'$ for every $C \succeq D$ and $I_N^{\succ} \not\models D'$.*

(v) *If for every clause $C \in N$, C is productive or $I_C \models C$, then $I_N^{\succ} \models N$.*

Proof. *(i) If C produces A, then $A \succeq L$ for every literal L of C. On the other hand, D contains $\neg A$, and $\neg A \succ A$. Since $\neg A \succ L$ for every literal L of C, we obtain $D \succ C$.*

(ii) Suppose that $I_D \models D$ and $C \succeq D$. If $I_D \models A$ for some positive literal A of D, then $A \in I_D \subseteq I_C \subseteq I_N^{\succ}$, so $I_C \models D$ and $I_N^{\succ} \models D$. Otherwise $I_D \models \neg A$ for some negative literal $\neg A$ of D, hence $A \notin I_D$. By (i), no clause that is larger than or equal to D produces A, so $A \notin I_C$ and $A \notin I_N^{\succ}$. Again, $I_C \models D$ and $I_N^{\succ} \models D$.

(iii) Obvious, since $C \succ D$ implies $A \in \Delta_D \subseteq I_C \subseteq I_N^{\succ}$.

(iv) If $D = D' \vee A$ produces A, then $A \succ L$ for every literal L of D' and $I_D \not\models A$. Since $I_D \not\models D$, we have $I_D \not\models L$ for every literal L of D'. Let $C \succeq D$. If L is a positive literal A', then $A' \notin I_D$. Since all atoms in $I_C \setminus I_D$ and $I_N^{\succ} \setminus I_D$ are larger than or equal to A, we get $A' \notin I_C$ and $A' \notin I_N^{\succ}$. Otherwise L is a negative literal $\neg A'$, then obviously $A' \in I_D \subseteq I_C \subseteq I_N^{\succ}$. In both cases L is false in I_C and $I_N^{\succ}$.

(v) By (ii) and (iii). $\square$

Model Existence Theorem

Proposition 3.18 *Let $\succ$ be a clause ordering. If N is saturated w. r. t. Res and $\bot \notin N$, then for every clause $C \in N$, C is productive or $I_C \models C$.*

Proof. Let N be saturated w. r. t. *Res* and $\bot \notin N$. Assume that the proposition does not hold. By well-foundedness, there must exist a minimal clause $C \in N$ (w. r. t. $\succ$) such that C is neither productive nor $I_C \models C$. As $C \neq \bot$ there exists a maximal literal in C. There are two possible reasons why C is not productive:

Case 1: The maximal literal $\neg A$ is negative, i. e., $C = C' \vee \neg A$. Then $I_C \models A$ and $I_C \not\models C'$. So some $D = D' \vee A \in N$ with $C \succ D$ produces A, and $I_C \not\models D'$. The inference

$$\frac{D' \vee A \qquad C' \vee \neg A}{D' \vee C'}$$

yields a clause $D' \vee C' \in N$ that is smaller than C. As $I_C \not\models D' \vee C'$, we know that $D' \vee C'$ is neither productive nor $I_{D' \vee C'} \models D' \vee C'$. This contradicts the minimality of C.

Case 2: The maximal literal A is positive, but not strictly maximal, i. e., $C = C' \vee A \vee A$. Then there is an inference

$$\frac{C' \vee A \vee A}{C' \vee A}$$

that yields a smaller clause $C' \vee A \in N$. As $I_C \not\models C' \vee A$, this clause is neither productive nor $I_{C' \vee A} \models C' \vee A$. Since $C \succ C' \vee A$, this contradicts the minimality of C. □

Theorem 3.19 (Bachmair & Ganzinger 1990) *Let $\succ$ be a clause ordering. If N is saturated w. r. t. Res and $\bot \notin N$, then $I_N^{\succ} \models N$.*

Proof. By Prop. 3.18 and part (v) of Prop. 3.17. □

Corollary 3.20 *Let N be saturated w. r. t. Res. Then $N \models \bot$ if and only if $\bot \in N$.*

Compactness of Propositional Logic

Lemma 3.21 *Let N be a set of propositional (or first-order ground) clauses. Then N is unsatisfiable, if and only if some finite subset $N' \subseteq N$ is unsatisfiable.*

Proof. The "if" part is trivial. For the "only if" part, assume that N be unsatisfiable. Consequently, $Res^*(N)$ is unsatisfiable as well. By refutational completeness of resolution, $\bot \in Res^*(N)$. So there exists an $n \geq 0$ such that $\bot \in Res^n(N)$, which means that $\bot$ has a finite resolution proof. Now choose N' as the set of assumptions in this proof. □

Theorem 3.22 (Compactness for Propositional Formulas) *Let S be a set of propositional (or first-order ground) formulas. Then S is unsatisfiable, if and only if some finite subset $S' \subseteq S$ is unsatisfiable.*

Proof. The "if" part is again trivial. For the "only if" part, assume that S be unsatisfiable. Transform S into an equivalent set N of clauses. By the previous lemma, N has a finite unsatisfiable subset N'. Now choose for every clause C in N' one formula F of S such that C is contained in the CNF of F. Let S' be the set of these formulas. □

3.11 General Resolution

Propositional (ground) resolution:

refutationally complete,

in its most naive version: not guaranteed to terminate for satisfiable sets of clauses, (improved versions do terminate, however)

inferior to the CDCL procedure.

But: in contrast to the CDCL procedure, resolution can be easily extended to non-ground clauses.

Observation

If $\mathcal{A}$ is a model of an (implicitly universally quantified) clause C, then by Lemma 3.8 it is also a model of all (implicitly universally quantified) instances $C\sigma$ of C.

Consequently, if we show that some instances of clauses in a set N are unsatisfiable, then we have also shown that N itself is unsatisfiable.

General Resolution through Instantiation

Idea: instantiate clauses appropriately:

$$
\begin{tikzcd}[column sep=small]
P(z',z') \vee \neg Q(z) \arrow[d, "{\{z' \mapsto a,\, z \mapsto f(a,b)\}}"'] & & & \neg P(a,y) \arrow[dl, "{\{y \mapsto a\}}"'] \arrow[dr, "{\{y \mapsto b\}}"] & & & P(x',b) \vee Q(f(x',x)) \arrow[d, "{\{x' \mapsto a,\, x \mapsto b\}}"] \\
P(a,a) \vee \neg Q(f(a,b)) \arrow[dr] & & \neg P(a,a) \arrow[dl] & & \neg P(a,b) \arrow[dr] & & P(a,b) \vee Q(f(a,b)) \arrow[dl] \\
& \neg Q(f(a,b)) \arrow[drr] & & & & Q(f(a,b)) \arrow[dll] & \\
& & & \bot & & &
\end{tikzcd}
$$

Early approaches (Gilmore 1960, Davis and Putnam 1960):

Generate ground instances of clauses.

Try to refute the set of ground instances by resolution.

If no contradiction is found, generate more ground instances.

Problems:

More than one instance of a clause can participate in a proof.

Even worse: There are infinitely many possible instances.

Observation:

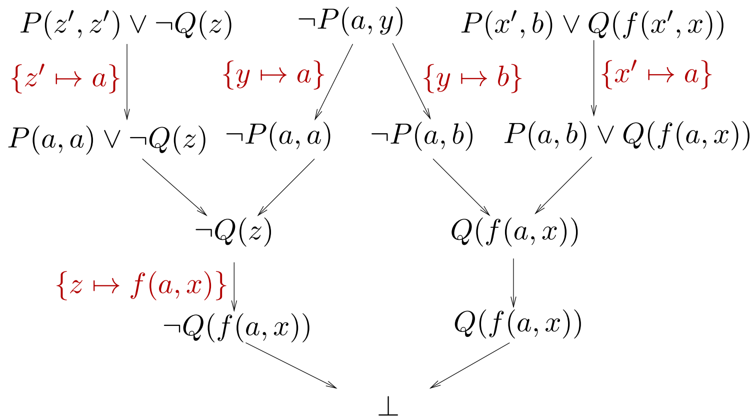
Instantiation must produce complementary literals (so that inferences become possible).

Idea (Robinson 1965):

Do not instantiate more than necessary to get complementary literals
$\Rightarrow$ most general unifiers (mgu).

Calculus works with non-ground clauses; inferences with non-ground clauses represent infinite sets of ground inferences which are computed simultaneously
$\Rightarrow$ lifting principle.

Computation of instances becomes a by-product of boolean reasoning.



Unification

Let $E = \{s_1 \doteq t_1, \ldots, s_n \doteq t_n\}$ (s_i, t_i terms or atoms) be a multiset of *equality problems*. A substitution σ is called a *unifier* of E if $s_i\sigma = t_i\sigma$ for all $1 \leq i \leq n$.

If a unifier of E exists, then E is called *unifiable*.

A substitution σ is called *more general* than a substitution τ, denoted by $\sigma \leq \tau$, if there exists a substitution ρ such that $\rho \circ \sigma = \tau$, where $(\rho \circ \sigma)(x) := (x\sigma)\rho$ is the composition of σ and ρ as mappings. (Note that $\rho \circ \sigma$ has a finite domain as required for a substitution.)

If a unifier of E is more general than any other unifier of E, then we speak of a *most general unifier* of E, denoted by $\mathrm{mgu}(E)$.

Proposition 3.23

(i) $\leq$ *is a quasi-ordering on substitutions, and* $\circ$ *is associative.*

(ii) If $\sigma \leq \tau$ *and* $\tau \leq \sigma$ *(we write* $\sigma \sim \tau$ *in this case), then* $x\sigma$ *and* $x\tau$ *are equal up to (bijective) variable renaming, for any* x *in* X*.*

A substitution σ is called *idempotent*, if $\sigma \circ \sigma = \sigma$.

Proposition 3.24 *σ is idempotent if and only if* $\mathrm{dom}(\sigma) \cap \mathrm{codom}(\sigma) = \emptyset$.

Rule-Based Naive Standard Unification

$$
\begin{array}{rcl}
t \doteq t, E & \Rightarrow_{SU} & E \\
f(s_1, \ldots, s_n) \doteq f(t_1, \ldots, t_n), E & \Rightarrow_{SU} & s_1 \doteq t_1, \ldots, s_n \doteq t_n, E \\
f(\ldots) \doteq g(\ldots), E & \Rightarrow_{SU} & \bot \\
 & & \text{if } f \neq g \\
x \doteq t, E & \Rightarrow_{SU} & x \doteq t, E\{x \mapsto t\} \\
 & & \text{if } x \in \mathrm{var}(E), x \notin \mathrm{var}(t) \\
x \doteq t, E & \Rightarrow_{SU} & \bot \\
 & & \text{if } x \neq t, x \in \mathrm{var}(t) \\
t \doteq x, E & \Rightarrow_{SU} & x \doteq t, E \\
 & & \text{if } t \notin X
\end{array}
$$

SU: Main Properties

If $E = \{x_1 \doteq u_1, \ldots, x_k \doteq u_k\}$, with x_i pairwise distinct, $x_i \notin \mathrm{var}(u_j)$, then E is called an (equational problem in) *solved form* representing the solution $\sigma_E = \{x_1 \mapsto u_1, \ldots, x_k \mapsto u_k\}$.

Proposition 3.25 *If E is a solved form then σ_E is an mgu of E.*

Theorem 3.26

1. *If* $E \Rightarrow_{SU} E'$ *then* σ *is a unifier of* E *if and only if* σ *is a unifier of* E'
2. *If* $E \Rightarrow^*_{SU} \bot$ *then* E *is not unifiable.*
3. *If* $E \Rightarrow^*_{SU} E'$ *with* E' *in solved form, then* $\sigma_{E'}$ *is an mgu of* E*.*

Proof. (1) We have to show this for each of the rules. Let's treat the case for the 4th rule here. Suppose σ is a unifier of $x \doteq t$, that is, $x\sigma = t\sigma$. Thus, $\sigma \circ \{x \mapsto t\} = \sigma[x \mapsto t\sigma] = \sigma[x \mapsto x\sigma] = \sigma$. Therefore, for any equation $u \doteq v$ in E: $u\sigma = v\sigma$, if and only if $u\{x \mapsto t\}\sigma = v\{x \mapsto t\}\sigma$. (2) and (3) follow by induction from (1) using Proposition 3.25. □

Main Unification Theorem

Theorem 3.27 *E is unifiable if and only if there is a most general unifier σ of E, such that σ is idempotent and* $\mathrm{dom}(\sigma) \cup \mathrm{codom}(\sigma) \subseteq \mathrm{var}(E)$.

Proof. The right-to-left implication is trivial. For the left-to-right implication we observe the following:

- $\Rightarrow_{SU}$ is terminating. A suitable lexicographic ordering on the multisets E (with $\bot$ minimal) shows this. Compare in this order:
  (1) the number of variables that occur in E below a function or predicate symbol, or on the right-hand side of an equation, or at least twice;
  (2) the multiset of the sizes (numbers of symbols) of all equations in E;
  (3) the number of non-variable left-hand sides of equations in E.
- A system E that is irreducible w. r. t. $\Rightarrow_{SU}$ is either $\bot$ or a solved form.
- Therefore, reducing any E by SU will end (no matter what reduction strategy we apply) in an irreducible E' having the same unifiers as E, and we can read off the mgu (or non-unifiability) of E from E' (Theorem 3.26, Proposition 3.25).
- σ is idempotent because of the substitution in rule 4. $\mathrm{dom}(\sigma) \cup \mathrm{codom}(\sigma) \subseteq \mathrm{var}(E)$, as no new variables are generated.

□

Rule-Based Polynomial Unification

Problem: Using $\Rightarrow_{SU}$, an *exponential growth* of terms is possible.

The following unification algorithm avoids this problem, at least if the final solved form is represented as a DAG.

$$
\begin{array}{rcl}
t \doteq t, E & \Rightarrow_{PU} & E \\
f(s_1, \ldots, s_n) \doteq f(t_1, \ldots, t_n), E & \Rightarrow_{PU} & s_1 \doteq t_1, \ldots, s_n \doteq t_n, E \\
f(\ldots) \doteq g(\ldots), E & \Rightarrow_{PU} & \bot \\
 & & \text{if } f \neq g \\
x \doteq y, E & \Rightarrow_{PU} & x \doteq y, E\{x \mapsto y\} \\
 & & \text{if } x \in \mathrm{var}(E), x \neq y \\
x_1 \doteq t_1, \ldots, x_n \doteq t_n, E & \Rightarrow_{PU} & \bot \\
 & & \text{if there are positions } p_i \text{ with} \\
 & & t_i|_{p_i} = x_{i+1}, t_n|_{p_n} = x_1 \\
 & & \text{and some } p_i \neq \varepsilon
\end{array}
$$

$$\begin{aligned}
x \doteq t, E \quad &\Rightarrow_{PU} \quad \bot \\
& \qquad\quad \text{if } x \neq t, x \in \text{var}(t) \\
t \doteq x, E \quad &\Rightarrow_{PU} \quad x \doteq t, E \\
& \qquad\quad \text{if } t \notin X \\
x \doteq t, x \doteq s, E \quad &\Rightarrow_{PU} \quad x \doteq t, t \doteq s, E \\
& \qquad\quad \text{if } t, s \notin X \text{ and } |t| \leq |s|
\end{aligned}$$

Properties of PU

Theorem 3.28

1. *If $E \Rightarrow_{PU} E'$ then σ is a unifier of E if and only if σ is a unifier of E'*
2. *If $E \Rightarrow^{*}_{PU} \bot$ then E is not unifiable.*
3. *If $E \Rightarrow^{*}_{PU} E'$ with E' in solved form, then $\sigma_{E'}$ is an mgu of E.*

Note: The solved form of $\Rightarrow_{PU}$ is different from the solved form obtained from $\Rightarrow_{SU}$. In order to obtain the unifier $\sigma_{E'}$, we have to sort the list of equality problems $x_i \doteq t_i$ in such a way that x_i does not occur in t_j for $j < i$, and then we have to compose the substitutions $\{x_1 \mapsto t_1\} \circ \cdots \circ \{x_k \mapsto t_k\}$.

Resolution for General Clauses

We obtain the resolution inference rules for non-ground clauses from the inference rules for ground clauses by replacing equality by unifiabilty:

General resolution *Res*:

$$\frac{D \vee B \qquad C \vee \neg A}{(D \vee C)\sigma} \quad \text{if } \sigma = \text{mgu}(A, B) \qquad \text{[resolution]}$$

$$\frac{C \vee A \vee B}{(C \vee A)\sigma} \quad \text{if } \sigma = \text{mgu}(A, B) \quad \text{[factorization]}$$

For inferences with more than one premise, we assume that the variables in the premises are (bijectively) renamed such that they become different to any variable in the other premises. We do not formalize this. Which names one uses for variables is otherwise irrelevant.

Lifting Lemma

Lemma 3.29 *Let C and D be variable-disjoint clauses. If*

$$\begin{array}{ccc} D & & C \\ \big\downarrow \theta_1 & & \big\downarrow \theta_2 \\ \end{array}$$

$$\frac{D\theta_1 \qquad C\theta_2}{C'} \qquad \textit{[ground resolution]}$$

then there exists a substitution ρ such that

$$\frac{D \qquad C}{C''} \qquad \textit{[general resolution]}$$

$$\big\downarrow \rho$$

$$C' = C''\rho$$

An analogous lifting lemma holds for factorization.

Saturation of Sets of General Clauses

Corollary 3.30 *Let N be a set of general clauses saturated under Res, i. e., $Res(N) \subseteq N$. Then also $G_\Sigma(N)$ is saturated, that is,*

$$Res(G_\Sigma(N)) \subseteq G_\Sigma(N).$$

Proof. W.l.o.g. we may assume that clauses in N are pairwise variable-disjoint. (Otherwise make them disjoint, and this renaming process changes neither $Res(N)$ nor $G_\Sigma(N)$.)

Let $C' \in Res(G_\Sigma(N))$. Then either (i) there exist resolvable ground instances $D\theta_1$ and $C\theta_2$ of N with resolvent C', or else (ii) C' is a factor of a ground instance $C\theta$ of C.

Case (i): By the Lifting Lemma, D and C are resolvable with a resolvent C'' with $C''\rho = C'$, for a suitable substitution ρ. As $C'' \in N$ by assumption, we obtain that $C' \in G_\Sigma(N)$.

Case (ii): Similar. $\square$

Soundness for General Clauses

Proposition 3.31 *The general resolution calculus is sound.*

Proof. We have to show that, if $\sigma = \mathrm{mgu}(A, B)$ then $\{\forall\vec{x}\ (D \vee B), \forall\vec{y}\ (C \vee \neg A)\} \models \forall\vec{z}\ (D \vee C)\sigma$ and $\{\forall\vec{x}\ (C \vee A \vee B)\} \models \forall\vec{z}\ (C \vee A)\sigma$.

Let $\mathcal{A}$ be a model of $\forall\vec{x}\ (D \vee B)$ and $\forall\vec{y}\ (C \vee \neg A)$. By Lemma 3.8, $\mathcal{A}$ is also a model of $\forall\vec{z}\ (D \vee B)\sigma$ and $\forall\vec{z}\ (C \vee \neg A)\sigma$ and by Lemma 3.7, $\mathcal{A}$ is also a model of $(D \vee B)\sigma$ and $(C \vee \neg A)\sigma$. Let β be an assignment. If $\mathcal{A}(\beta)(B\sigma) = 0$, then $\mathcal{A}(\beta)(D\sigma) = 1$. Otherwise $\mathcal{A}(\beta)(B\sigma) = \mathcal{A}(\beta)(A\sigma) = 1$, hence $\mathcal{A}(\beta)(\neg A\sigma) = 0$ and therefore $\mathcal{A}(\beta)(C\sigma) = 1$. In both cases $\mathcal{A}(\beta)((D \vee C)\sigma) = 1$, so $\mathcal{A} \models (D \vee C)\sigma$ and by Lemma 3.7, $\mathcal{A} \models \forall\vec{z}\ (D \vee C)\sigma$.

The proof for factorization inferences is similar. $\square$

Herbrand's Theorem

Lemma 3.32 *Let N be a set of Σ-clauses, let $\mathcal{A}$ be an interpretation. Then $\mathcal{A} \models N$ implies $\mathcal{A} \models G_\Sigma(N)$.*

Lemma 3.33 *Let N be a set of Σ-clauses, let $\mathcal{A}$ be an* Herbrand *interpretation. Then $\mathcal{A} \models G_\Sigma(N)$ implies $\mathcal{A} \models N$.*

Proof. Let $\mathcal{A}$ be an Herbrand model of $G_\Sigma(N)$. We have to show that $\mathcal{A} \models \forall\vec{x}\ C$ for all clauses $\forall\vec{x}\ C$ in N. This is equivalent to $\mathcal{A} \models C$, which in turn is equivalent to $\mathcal{A}(\beta)(C) = 1$ for all assignments β.

Choose $\beta : X \to U_{\mathcal{A}}$ arbitrarily. Since $\mathcal{A}$ is an Herbrand interpretation, $\beta(x)$ is a ground term for every variable x, so there is a substitution σ such that $x\sigma = \beta(x)$ for all variables x occurring in C. Now let γ be an arbitrary assignment, then for every variable occurring in C we have $(\gamma \circ \sigma)(x) = \mathcal{A}(\gamma)(x\sigma) = x\sigma = \beta(x)$ and consequently $\mathcal{A}(\beta)(C) = \mathcal{A}(\gamma \circ \sigma)(C) = \mathcal{A}(\gamma)(C\sigma)$. Since $C\sigma \in G_\Sigma(N)$ and $\mathcal{A}$ is an Herbrand model of $G_\Sigma(N)$, we get $\mathcal{A}(\gamma)(C\sigma) = 1$, so $\mathcal{A}$ is a model of C. $\square$

Theorem 3.34 (Herbrand) *A set N of Σ-clauses is satisfiable if and only if it has an Herbrand model over Σ.*

Proof. The "$\Leftarrow$" part is trivial. For the "$\Rightarrow$" part let $N \not\models \bot$. Since resolution is sound, this implies that $\bot \notin \mathit{Res}^*(N)$. Obviously, a ground instance of a clause has the same number of literals as the clause itself, so we can conclude that $\bot \notin G_\Sigma(\mathit{Res}^*(N))$. Since $\mathit{Res}^*(N)$ is saturated, $G_\Sigma(\mathit{Res}^*(N))$ is saturated as well by Cor. 3.30. Now $I_{G_\Sigma(\mathit{Res}^*(N))}$ is an Herbrand interpretation over Σ and by Thm. 3.19 it is a model of $G_\Sigma(\mathit{Res}^*(N))$. By Lemma 3.33, every Herbrand model of $G_\Sigma(\mathit{Res}^*(N))$ is a model of $\mathit{Res}^*(N)$. Now $N \subseteq \mathit{Res}^*(N)$, so $I_{G_\Sigma(\mathit{Res}^*(N))} \models N$. $\square$

Corollary 3.35 *A set N of Σ-clauses is satisfiable if and only if its set of ground instances $G_\Sigma(N)$ is satisfiable.*

Proof. The "$\Rightarrow$" part follows directly from Lemma 3.32. For the "$\Leftarrow$" part assume that $G_\Sigma(N)$ is satisfiable. By Thm. 3.34 $G_\Sigma(N)$ has an Herbrand model. By Lemma 3.33, every Herbrand model of $G_\Sigma(N)$ is a model of N. $\square$

Refutational Completeness of General Resolution

Theorem 3.36 *Let N be a set of general clauses that is saturated w. r. t. Res. Then $N \models \bot$ if and only if $\bot \in N$.*

Proof. The "$\Leftarrow$" part is trivial. For the "$\Rightarrow$" part assume that N is saturated, that is, $Res(N) \subseteq N$. By Corollary 3.30, $G_\Sigma(N)$ is saturated as well, i. e., $Res(G_\Sigma(N)) \subseteq G_\Sigma(N)$. By Cor. 3.35, $N \models \bot$ implies $G_\Sigma(N) \models \bot$. By the refutational completeness of ground resolution, $G_\Sigma(N) \models \bot$ implies $\bot \in G_\Sigma(N)$, so $\bot \in N$. $\square$

3.12 Theoretical Consequences

We get some classical results on properties of first-order logic as easy corollaries.

The Theorem of Löwenheim-Skolem

Theorem 3.37 (Löwenheim–Skolem) *Let Σ be a countable signature and let S be a set of closed Σ-formulas. Then S is satisfiable if and only if S has a model over a countable universe.*

Proof. If both X and Σ are countable, then S can be at most countably infinite. Now generate, maintaining satisfiability, a set N of clauses from S. This extends Σ by at most countably many new Skolem functions to Σ'. As Σ' is countable, so is $T_{\Sigma'}$, the universe of Herbrand-interpretations over Σ'. Now apply Theorem 3.34. $\square$

There exist more refined versions of this theorem. For instance, one can show that, if S has some infinite model, then S has a model with a universe of cardinality κ for every κ that is larger than or equal to the cardinalty of the signature Σ.

Compactness of Predicate Logic

Theorem 3.38 (Compactness Theorem for First-Order Logic) *Let S be a set of closed first-order formulas. S is unsatisfiable $\Leftrightarrow$ some finite subset $S' \subseteq S$ is unsatisfiable.*

Proof. The "$\Leftarrow$" part is trivial. For the "$\Rightarrow$" part let S be unsatisfiable and let N be the set of clauses obtained by Skolemization and CNF transformation of the formulas in S. Clearly $Res^*(N)$ is unsatisfiable. By Theorem 3.36, $\bot \in Res^*(N)$, and therefore $\bot \in Res^n(N)$ for some $n \in \mathbb{N}$. Consequently, $\bot$ has a finite resolution proof B of depth $\leq n$. Choose S' as the subset of formulas in S such that the corresponding clauses contain the assumptions (leaves) of B. $\square$