3.5 Restrictions of Resolution

Dienstag, 16. Mai 2017 09:30

Plan:

3.5.1. Linear Resolution — maintains completeness

3.5.2. Input Resolution — not complete on arbitrary clauses, but complete on Horn clauses (LP only uses Horn clauses)

SLD Resolution — complete on Horn clauses

3.5.1. Linear Resolution

Def 3.5.1 (Linear Resolution) — Slide 19

Let $\mathcal{K}$ be a clause set.

$\square$ is derivable from a clause K in $\mathcal{K}$ by <u>linear resolution</u>

iff there exists a sequence $K_1, \dots, K_m$ with

$K_1 = K \in \mathcal{K}$ and $K_m = \square$

and for all $2 \leq i \leq m$ we have:

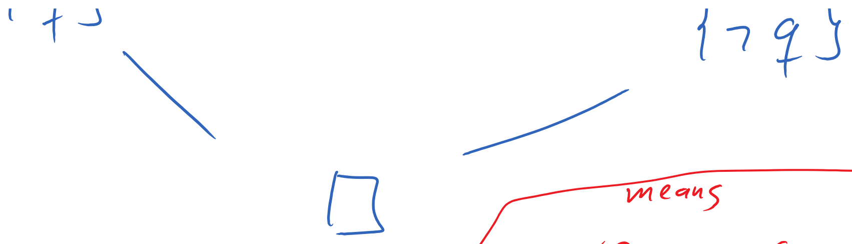
K_i is resolvent of K_{i-1} and a clause from $\{K_1, \dots, K_{i-1}\} \cup \mathcal{K}$.

Ex. 3.5.2 $p, q \in \Delta_0$

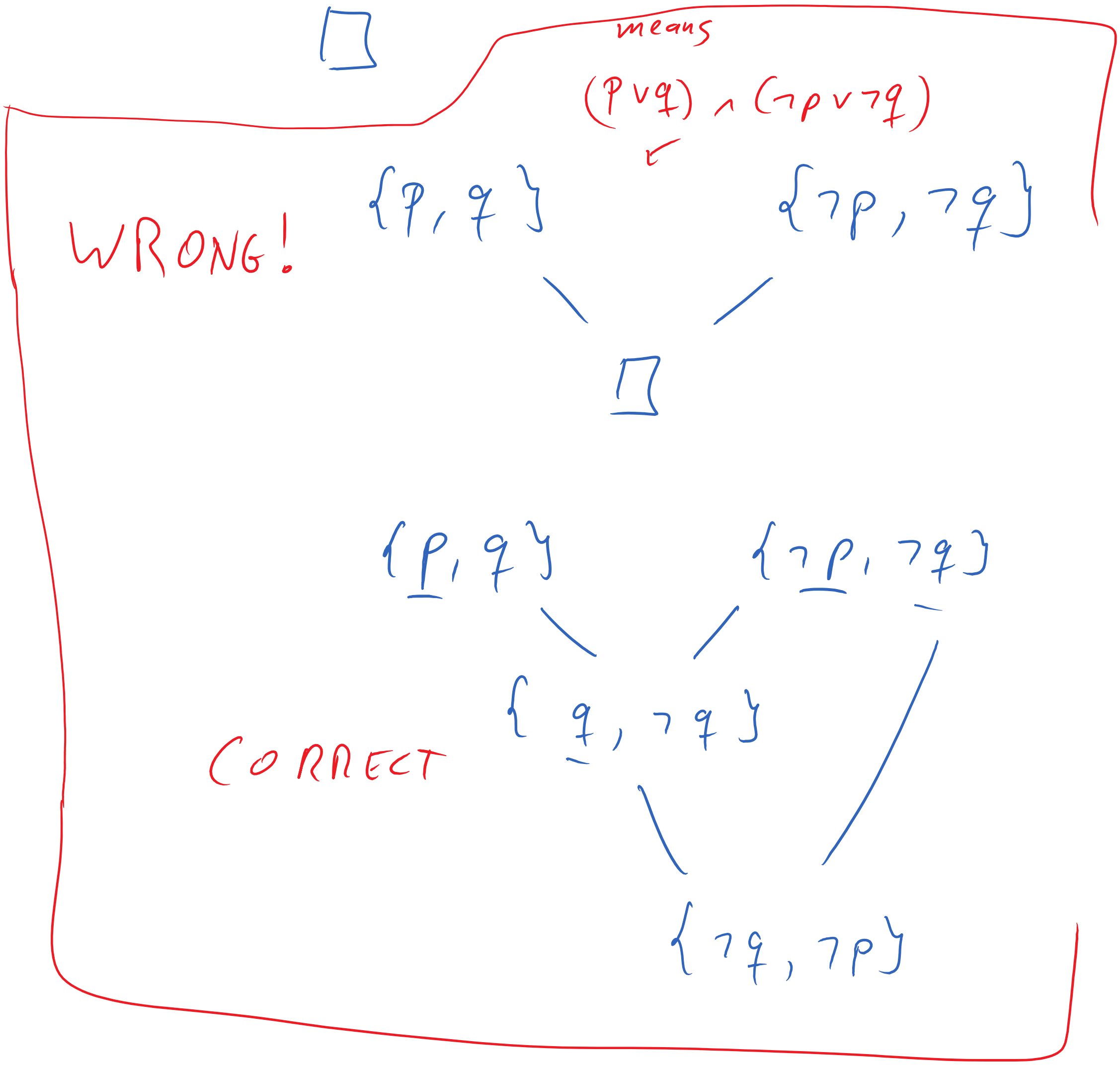
$\{p, q\}$ $\{\neg p, q\}$ $\{p, \neg q\}$ $\{\neg p, \neg q\}$

$\{q\}$ $\{\neg q\}$

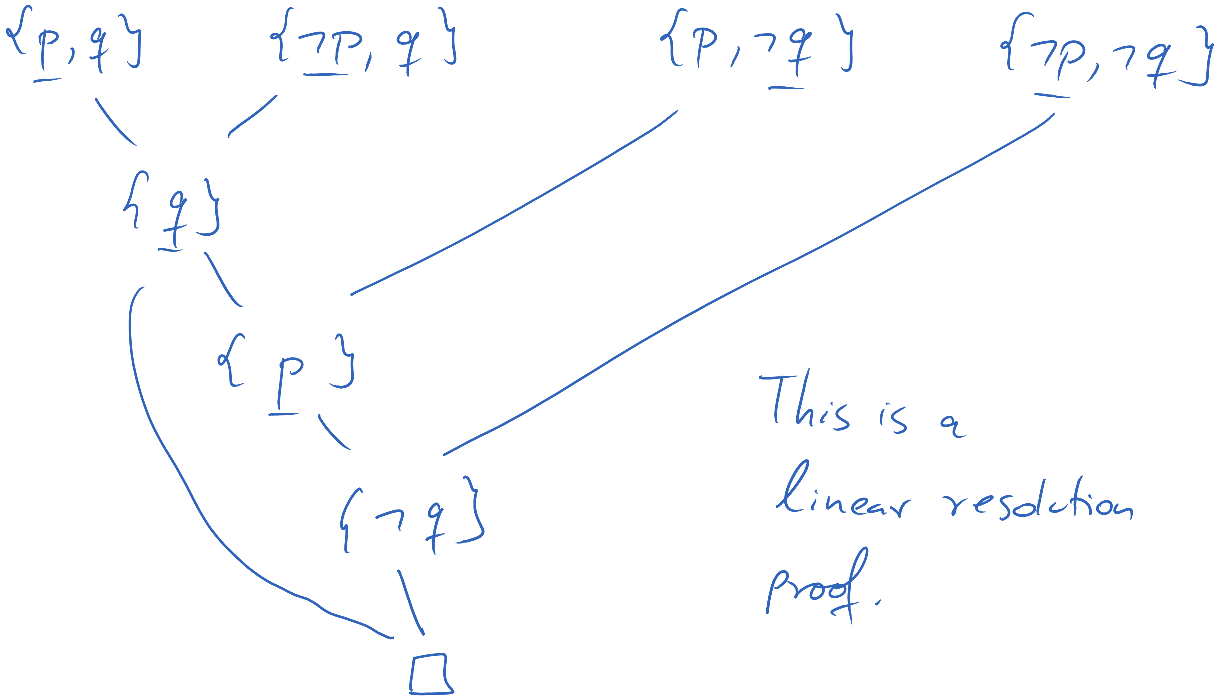
is not



is not a linear resolution proof



Can we also obtain □ by linear resolution?



Slide 19

This is a linear resolution proof.

Thm 3.5.3 (Soundness + Completeness of Linear Resolution)

Let $\mathcal{K}$ be a clause set.
Then $\mathcal{K}$ is unsatisfiable iff $\square$ can be derived by linear resolution from some $K \in \mathcal{K}$.
If $\mathcal{K}$ is a <u>minimal</u> unsatisfiable clause set, (i.e., every $\mathcal{K}' \subsetneq \mathcal{K}$ is satisfiable) then $\square$ can be derived by linear resolution from every $K \in \mathcal{K}$.

<u>Proof</u>: Soundness follows from Thm 3.4.10 (soundness of full resolution), because every linear resolution step is a resolution step.

For completeness, one first shows completeness of linear ground resolution. Then the lifting lemma is used to prove completeness of linear res. in pred. logic.
(Course Notes) $\boxed{\text{q}}$

3.5.2. Input Resolution and SLD Resolution

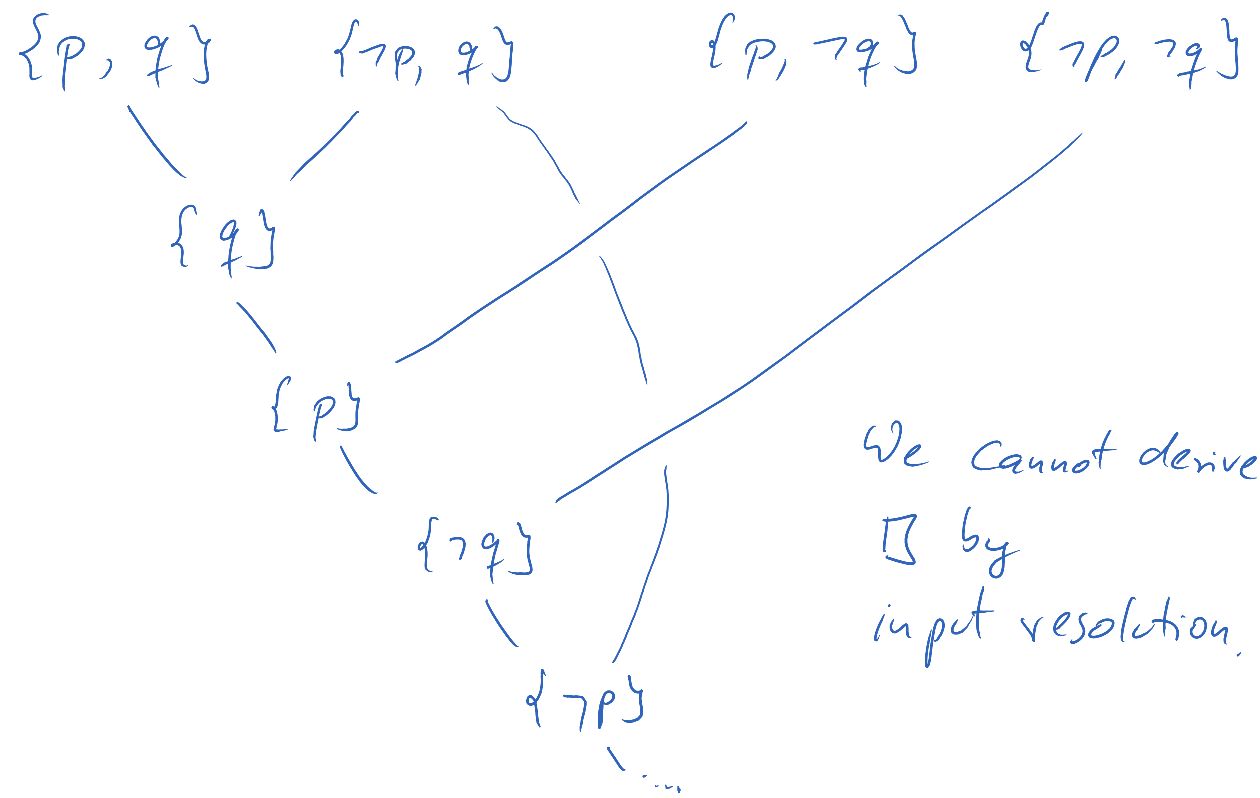
Idea: Restrict linear resolution further. The second parent clause should come from the original input clause set $\mathcal{K}$.

Def 354 (Input Resolution)

Let $\mathcal{K}$ be a clause set. Then $\Box$ can be derived from $K \in \mathcal{K}$ by <u>input resolution</u> iff there is a sequence $K_1, \dots, K_m$ where $K_1 = K \in \mathcal{K}$, $K_m = \Box$, and for all $2 \le i \le m$ we have:

K_i is a resolvent of K_{i-1} and a clause from $\mathcal{K}$.

Ex. 355 Is input resolution still complete?



We cannot derive $\Box$ by input resolution.

Thus: Input resolution is sound, but not complete.

Solution: Do not regard arbitrary clauses anymore, just regard <u>Horn clauses</u>.

It will turn out that on Horn clauses, input resolution is complete.

Def 356 (Horn clause)

A clause K is a Horn clause iff it contains at most one positive literal (i.e., at most one of its literals is an atomic formula and the other literals are negated atomic formulas).

A Horn clause is negative iff it only contains negative literals, i.e., it has the form $\{\neg A_1, \dots, \neg A_k\}$ for atomic formulas $A_1, \dots, A_k$.

A Horn clause is definite iff it contains a positive literal, i.e., it has the form $\{B, \neg C_1, \dots, \neg C_n\}$ for atomic formulas $B, C_1, \dots, C_n$.

A set of definite Horn clauses corresponds to a conjunction of implications:

$p, q, r, s \in \Delta_0$

$$\{ \ \{p, \neg q\}, \ \{\neg r, \neg p, s\}, \ \{s\} \ \}$$

is equivalent to

$$(p \vee \neg q) \wedge (\neg r \vee \neg p \vee s) \wedge s$$

which is equivalent to

$$(q \rightarrow p) \wedge (r \wedge p \rightarrow s) \wedge s$$

Connection to Logic Programming:

- Facts $\quad s.$

  $\hat{=}$ definite Horn clause $\{s\}$ without negative literals

- Rules $\quad s :- r, p. \quad (\hat{=} (r \wedge p) \to s \iff \neg(r \wedge p) \vee s$
  $\iff \neg r \vee \neg p \vee s)$

  $\hat{=}$ definite Horn clause $\{s, \neg r, \neg p\}$ with negative literals.

$s :- \neg r$
$\{s, r\}$

$\Big\}$ not allowed in pure logic programming.

Prolog has a form of negation $\Rightarrow$ Chapter 5

Even with pure logic prog., one can have infinitely many answers:

nat(0).
nat(s(X)) :- nat(X).

$\{nat(s(X)), \neg nat(X)\}$

?- nat(Y).

Y=0;
Y=s(0);
Y=s(s(0));

- Queries $\quad$?- p, q. $\quad$ "Does $p \wedge q$ hold?"

$\Rightarrow$ Negation yields $\neg p \vee \neg q$

Corresponds to negative Horn clause $\{\neg p, \neg q\}$

Restriction to Horn clauses affects the expressivity: $\{p, q\}$ has no equivalent Horn clause.

But this restriction improves efficiency:

- Propositional logic:
  - Satisfiability in general: decidable, NP-complete
  - Satisfiability for Horn Clauses: polynomial time
- Pred. Logic
  - Satisfiability in general: undecidable, input res. incomplete
  - Satisf. for Horn clauses: undecidability, input res. complete

If we restrict ourselves to Horn clauses, then input resolution can be improved further to SLD resolution. We will prove that SLD resolution is complete on Horn clauses (which implies that input resol. is complete on Horn clauses).

Idea: Input resolution proofs should start with a negative Horn clause.

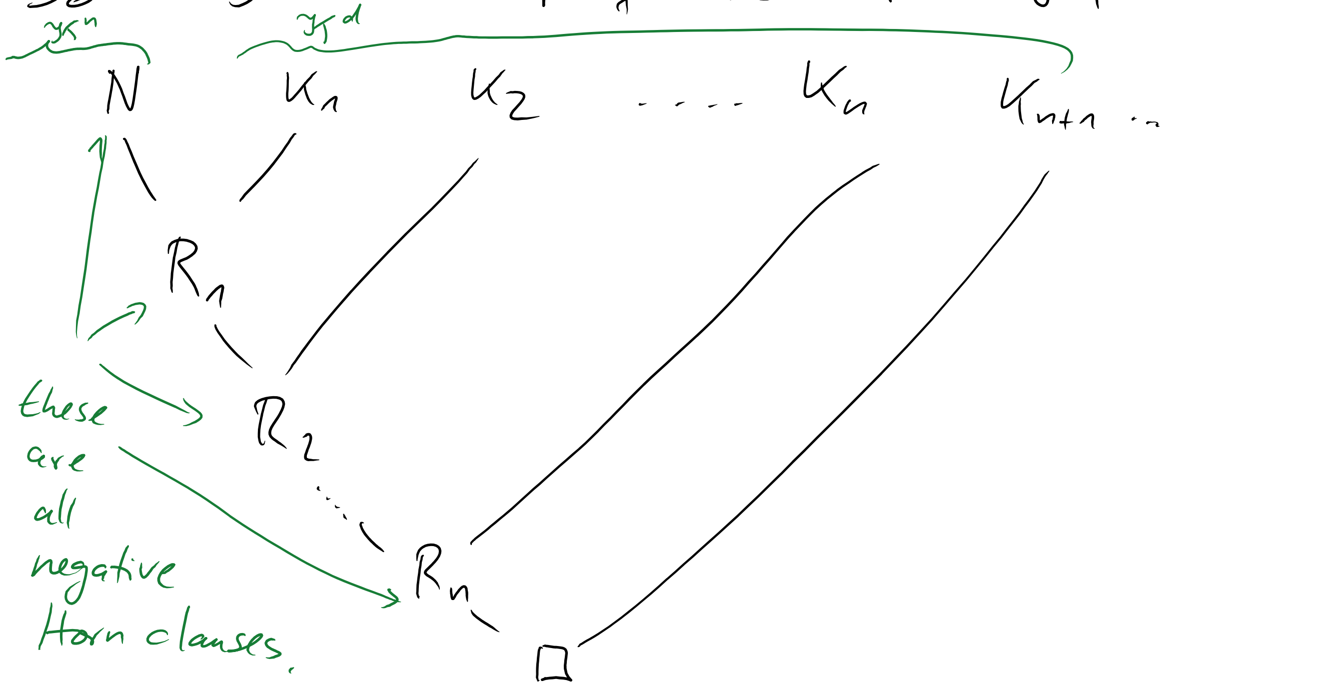
Def 357 (SLD Resolution)

Let $\mathcal{K}$ be a set of Horn clauses with $\mathcal{K} = \mathcal{K}^d \,\dot\cup\, \mathcal{K}^n$ where $\mathcal{K}^d$ contains the definite Horn clauses from $\mathcal{K}$ and $\mathcal{K}^n$ contains the negative Horn clauses from $\mathcal{K}$.

Then $\Box$ can be derived from $K \in \mathcal{K}^n$ by SLD resolution iff there is a seq. $K_1, .., K_m$ with $K_1 = K \in \mathcal{K}^n$, $K_m = \Box$, and forall $2 \le i \le m$ we have:

K_i is a resolvent of K_{i-1} and a clause from $\mathcal{K}^d$.

So SLD resolution proofs have the following form:



A negative Horn clause can only be resolved with a definite Horn clause.

selects the next literal in the negative clause that should be used for resolution

"SLD": Linear resolution with selection function for <u>de</u>finite clauses

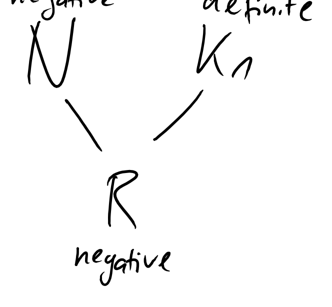
<u>Thm 3.5.8</u> (Soundness and Completeness of SLD Resolution)

Let $\mathcal{K}$ be a set of Horn clauses.
Then $\mathcal{K}$ is unsatisfiable iff $\Box$ can be derived from some negative clause $N \in \mathcal{K}$ by SLD resolution.

<u>Proof</u>: Soundness ("$\Leftarrow$") follows from Thm 3.4.10 (soundness of resolution).

Completeness ("$\Rightarrow$"): Let $\mathcal{K}_{min}$ be a minimal unsatisfiable subset of $\mathcal{K}$. Every set of definite Horn clauses is satisfiable (by the interpretation that satisfies all atomic formulas). Thus, $\mathcal{K}_{min}$ contains a negative Horn clause N.

By Thm 3.5.3 (Completeness of linear resolution), $\Box$ can be derived by linear resolution from <u>every</u> clause of $\mathcal{K}_{min}$. Hence: there is a linear resolution proof of $\Box$ that starts with N.



- There cannot be a resolution step with 2 negative parent clauses.
- If one performs resolution with a negative and a definite clause,

then the resolvent is a negative clause.

$\Rightarrow$ This linear resolution proof of $\Box$ is an SLD-proof. $\boxtimes$

Logic programming uses one more restriction:

Instead of resolving $L_1, \dots, L_m$ from $\nu(K_1)$

and $L_1', \dots, L_n'$ from $\nu(K_2)$,

one fixes $m = n = 1$.

This is called <u>binary resolution</u>.

Binary resolution is not complete on arbitrary clauses!

<u>Ex. 3.5.9</u>

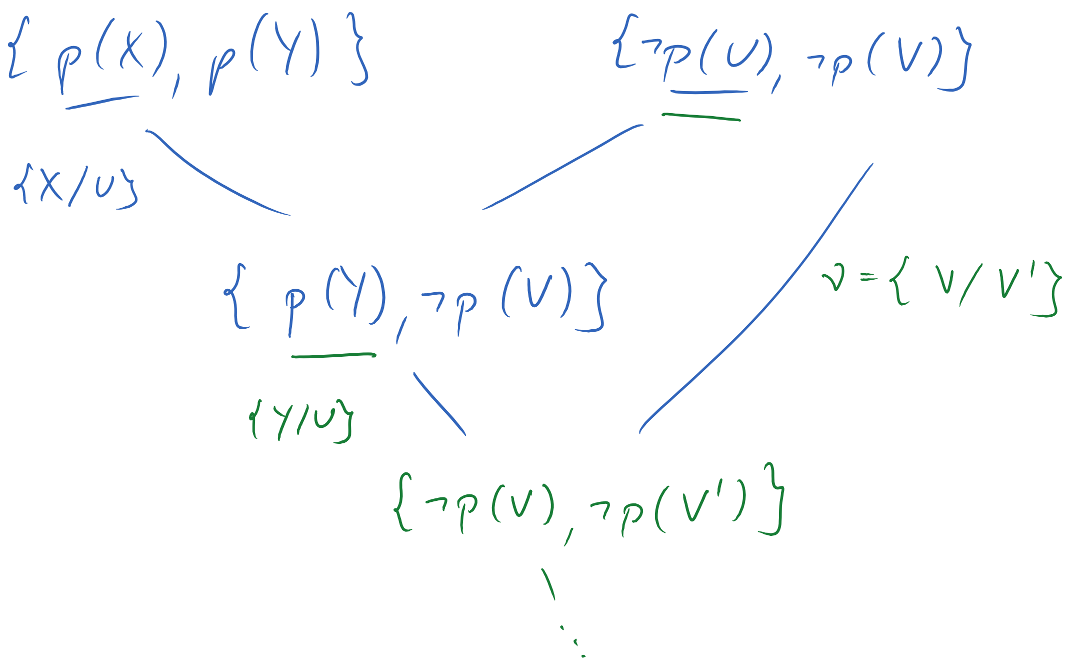
$\mathcal{K} = \{ \{p(X), p(Y)\}, \quad \{\neg p(U), \neg p(V)\} \}$

because $\Box$

$\{p(X), p(Y), p(U), p(V)\}$
are unifiable by
mgu $\{X/V, Y/V, U/V\}$.

Binary resolutions

$\{p(X), p(Y)\} \qquad \{\neg p(U), \neg p(V)\}$



One will never reach $\square$.

However, binary resolution is complete on Horn clauses.

Thm 3.5.10. (Soundness + Completeness of binary SLD Resolution)

Let $\mathcal{K}$ be a set of Horn clauses.
Then $\mathcal{K}$ is unsatisfiable iff $\square$ can be derived from a negative clause $N \in \mathcal{K}$ by binary SLD resolution.

Proof: Course Notes.