Automated Theorem Proving

Georg Moser

Institute of Computer Science @ UIBK

Winter 2015

Summary of Last Lecture

Definition

- let A be closed and rectified
- we define the mapping rsk as follows:

$$\text{rsk}(A) = \begin{cases} A & \text{no existential quant. in } A \\ \text{rsk}(A_{-\exists y})\{y \mapsto f(x_1, \ldots, x_n)\} & \forall x_1, \ldots, \forall x_n <_A \exists y \end{cases}$$

1. $\exists y$ is the first existential quantifier in A
2. $A_{-\exists y}$ denotes A after omission of $\exists y$
3. the Skolem function symbol f is fresh

- the formula rsk(A) is the (refutational) structural Skolem form of A

Inner and Outer (Refutational) Skolemisation

Definition

- let A be rectified sentence in negation normal form (NNF)
- let $\exists x B$ a subformula of A at position p
- let $\{y_1, \ldots, y_k\} = \{y \mid \forall y <_A \exists x\}$ and let $\{z_1, \ldots, z_l\} = \mathcal{FV}\text{ar}(\exists x B)$
- $A[B\{x \mapsto f(y_1, \ldots, y_k)\}]$ is obtained by an outer Skolemisation step
- $A[B\{x \mapsto f(z_1, \ldots, z_l)\}]$ is obtained by an inner Skolemisation step

Example

1. structural Skolemisation is a variation of outer Skolemisation
2. Andrew's Skolemisation is a variation of inner and outer Skolemisation

Outline of the Lecture

Early Approaches in Automated Reasoning

Herbrand's theorem for dummies, Gilmore's prover, method of Davis and Putnam

Starting Points

resolution, tableau provers, Skolemisation, ordered resolution, redundancy and deletion

Automated Reasoning with Equality

paramodulation, ordered completion and proof orders, superposition

Applications of Automated Reasoning

Neuman-Stubblebinde Key Exchange Protocol, Robbins problem

Definition (Optimised Skolemisation)

- let A be a sentence in NNF and $B = \exists x_1 \cdots \exists x_k (E \wedge F)$ a subformula of A with $\mathcal{FV}\mathrm{ar}(\exists \vec{x}(E \wedge F)) = \{y_1, \dots, y_n\}$
- suppose $A = C[B]$
- suppose $A \to \forall y_1, \dots, \forall y_n \exists x_1 \cdots \exists x_k E$ is valid
- we define an optimised Skolemisation step as follows

  $$\mathrm{opt\_step}(A) = \forall \vec{y} E\{\dots, x_i \mapsto f_i(\vec{y}), \dots\} \wedge C[F\{\dots, x_i \mapsto f_i(\vec{y}), \dots\}]$$

  where $f_1, \dots, f_k$ are new Skolem function symbols

Example

consider a subformula of a sentence A

$$\forall x \forall y \forall z (\mathrm{R}(x,y) \wedge \mathrm{R}(x,z) \to \exists u(\mathrm{R}(y,u) \wedge \mathrm{R}(z,u)))$$

we exemplarily assume $\forall y \exists u \mathrm{R}(y,u)$ is provable from A; we obtain

$$\mathrm{R}(y, \mathrm{f}(y,z)) \qquad \neg\mathrm{R}(x,y) \vee \neg\mathrm{R}(x,z) \vee \mathrm{R}(z, \mathrm{f}(y,z))$$

Theorem

optimised Skolemisation preserves satisfiability

Proof Sketch.

1. suppose A is satisfiable with some interpretation $\mathcal{I}$
2. we extent $\mathcal{I}$ to the Skolem functions such that we obtain for the extention $\mathcal{I}'$

   $$\mathcal{I}' \models \forall \vec{y} E\{\dots, x_i \mapsto f_i(\vec{y}), \dots\} \qquad \mathcal{I}' \models C[F\{\dots, x_i \mapsto f_i(\vec{y}), \dots\}]$$

3. for this the extra condition is exploited

■

Remark

note that in optimised Skolemisation some literals are deleted from clauses

Definition

- a clause C subsumes clause D, if $\exists\ \sigma$ such that the multiset of literals of $C\sigma$ is contained in the multiset of literals of D (denoted $C\sigma \subseteq D$)
- C is a condensation of D if C is a proper (multiple) factor of D that subsumes D

Example

consider the clause $\mathrm{P}(x) \vee \mathrm{R}(\mathrm{b}) \vee \mathrm{P}(\mathrm{a}) \vee \mathrm{R}(z)$; its condensation is $\mathrm{R}(\mathrm{b}) \vee \mathrm{P}(\mathrm{a})$

NB: condensation forms a strong normalisation technique that is essential to remove redundancy in clauses

Example

note that the clause $\mathrm{R}(x,x) \vee \mathrm{R}(y,y)$ does not subsume $\mathrm{R}(\mathrm{a},\mathrm{a})$

Definition

- let $B = \exists \vec{x}(E_1 \wedge \cdots \wedge E_\ell)$ be a formula
- let $\{\vec{z}_1\} = \mathcal{FV}\mathrm{ar}(E_1) \setminus \{\vec{x}\}$
- let $\{\vec{z}_i\} = \mathcal{FV}\mathrm{ar}(E_i) \setminus \left(\bigcup_{j<i} \mathcal{FV}\mathrm{ar}(E_j) \cup \{\vec{x}\}\right)$
- we call $\langle \{\vec{z}_1\}, \dots, \{\vec{z}_\ell\} \rangle$ the (free variable) splitting of B

Example

consider $\exists u(\mathrm{R}(y,u) \wedge \mathrm{R}(z,u))$; its splitting is $\langle \{y\}, \{z\} \rangle$

Observation

- let $\langle \{\vec{z}_1\}, \dots, \{\vec{z}_\ell\} \rangle$ be a splitting of $\exists \vec{x}(E_1 \wedge \cdots \wedge E_\ell)$
- assume each conjunct E_i contains at least one of the variables from $\vec{x}$
- $\langle \{\vec{z}_1, \vec{z}_2\}, \dots, \{\vec{z}_\ell\} \rangle$ is a splitting of $\exists \vec{v}(E_2 \wedge \cdots \wedge E_\ell)\{x_i \mapsto f_i(\vec{z}_1, \vec{v})\}$ where $\vec{v}$ are new

Definition (Strong Skolemisation)

- let A be a sentence in NNF and $B = \exists \vec{x}(E_1 \wedge \cdots \wedge E_\ell)$ a subformula such that $A = C[B]$
- let $\langle\{\vec{z}_1\}, \dots, \{\vec{z}_\ell\}\rangle$ be a free variable splitting of B
- a strong Skolemisation step is defined as $\mathrm{str\_step}(A) = C[D]$ where D is defined as

$$\forall \vec{w}_2, \dots, \vec{w}_\ell E_1\{x_i \mapsto f_i(\vec{z}_1, \vec{w}_2, \dots, \vec{w}_\ell)\} \wedge \cdots$$
$$\cdots \wedge E_\ell\{x_i \mapsto f_i(\vec{z}_1, \vec{z}_2, \dots, \vec{z}_\ell)\}$$

Example

consider the formula $\forall x \forall y \forall z(\mathsf{R}(x,y) \wedge \mathsf{R}(x,z) \to \exists u(\mathsf{R}(y,u) \wedge \mathsf{R}(z,u)))$
strong Skolemisation yields the following clauses

$\neg\mathsf{R}(x,y) \vee \neg\mathsf{R}(x,z) \vee \mathsf{R}(y,\mathsf{f}(y,w))$ $\quad$ $\neg\mathsf{R}(x,y) \vee \neg\mathsf{R}(x,z) \vee \mathsf{R}(z,\mathsf{f}(y,z))$

condensation of the first clause yields: $\neg\mathsf{R}(x,y) \vee \mathsf{R}(y,\mathsf{f}(y,w))$

Lemma

if $\exists x_1 \dots \exists x_k(E \wedge F)$ is satisfiable, then the following formula is satisfiable as well

$$\forall w_1 \dots \forall w_k E\{x_i \mapsto f_i(\vec{y}, \vec{w})\} \wedge \exists v_1 \dots \forall v_k F\{x_i \mapsto f_i(\vec{y}, \vec{v})\}$$

where $\{y_1, \dots, y_n\} = \mathcal{FV}\mathrm{ar}(E) \setminus \{x_1, \dots, x_k\}$

Theorem

strong Skolemisation preserves satisfiability

Proof Sketch.

- suppose A is satisfiable
- one shows satisfiability of $\mathrm{str\_step}(A)$ by main induction on A and side induction on ℓ
- the base case exploits the above lemma ∎

Assessment

structural Skolemisation

- structural (outer) Skolemisation can lead to non-elementary speed-up over prenex Skolemisation
- structural Skolemisation requires non-trivial formula transformations, in particular quantifier shiftings
- how to implement?

inner Skolemisation

- standard inner Skolemisation techniques are straightforward to implement
- optimised Skolemisation requires proof of $A \to \forall \vec{y} \exists \vec{x} E$ as pre-condition
- strong Skolemisation is incomparable to optimised Skolemisation, as larger, but more general clauses may be produced

Definitions

- a proper order is a irreflexive and transitive relation
- a quasi-order is reflexive and transitive
- a partial order is an anti-symmetric quasi-order
- a proper order $\succ$ on a set A is well-founded (on A) if

$$\neg\exists\ a_1 \succ a_2 \succ \cdots \qquad a_i \in A$$

- a well-founded order is a well-founded proper order
- a linear (or total) order fulfills:
  $\forall\ a, b \in A$, $a \neq b$, either $a \succ b$ or $b \succ a$
- a well-order is a linear well-founded order

Example

$\geqslant$ on $\mathbb{N}$ is a partial order; we often write $(\mathbb{N}, \geqslant)$ to indicate the domain;
$(\mathbb{N}, \geqslant)$ is not well-founded, but $(\mathbb{N}, >)$ is a well-order

Orders on Literals

Definition

- let $\succ$ be a well-founded and total order on ground atomic formulas
- extend $\succ$ to a well-founded proper order $\succ_{\mathsf{L}}$ total on ground literals such that:
  1. if $A \succ B$, then $A \succ_{\mathsf{L}} B$ and $\neg A \succ_{\mathsf{L}} \neg B$
  2. $\neg A \succ_{\mathsf{L}} A$

Example

- consider a well-founded proper order $\succ$ on atoms that is total on ground atomic formulas
- identify an atom A with the multiset $\{A\}$ and $\neg A$ with $\{A, A\}$
- set $\succ_{\mathsf{L}} = \succ^{\mathrm{mul}}$
- $\succ_{\mathsf{L}}$ fulfills the above conditions

Ordered Resolution Calculus

Definition

σ is ground if $E\sigma$ is ground

- a literal L is maximal if $\exists$ ground σ such that for no other literal M: $M\sigma \succ_{\mathsf{L}} L\sigma$
- L is strictly maximal if $\exists$ ground σ such that for no other literal M: $M\sigma \succcurlyeq_{\mathsf{L}} L\sigma$; here $\succcurlyeq_{\mathsf{L}}$ denotes the reflexive closure

Definition

ordered resolution

$$\frac{C \lor A \quad D \lor \neg B}{(C \lor D)\sigma}$$

ordered factoring

$$\frac{C \lor A \lor B}{(C \lor A)\sigma}$$

1. σ is a mgu of the atomic formulas A and B
2. $A\sigma$ is strictly maximal with respect to $C\sigma$; $\neg B\sigma$ is maximal with respect to $D\sigma$

Example

consider the clause set (constants a, b, predicates P, Q, R, S)

$$\mathsf{P}(x) \lor \mathsf{Q}(x) \lor \mathsf{R}(x,y) \qquad \neg\mathsf{P}(x) \qquad \neg\mathsf{Q}(\mathsf{a})$$
$$\mathsf{S}(\mathsf{a},y) \lor \neg\mathsf{R}(\mathsf{a},y) \lor \mathsf{S}(x,\mathsf{b}) \qquad \neg\mathsf{S}(\mathsf{a},\mathsf{b}) \lor \neg\mathsf{R}(\mathsf{a},\mathsf{b})$$

together with the atom order $\mathsf{P}(t_1) \succ \mathsf{Q}(t_2) \succ \mathsf{S}(t_3, t_4) \succ \mathsf{R}(t_5, t_6)$

$$\sqcap \qquad \cfrac{\cfrac{\mathsf{P}(x) \lor \mathsf{Q}(x) \lor \mathsf{R}(x,y) \quad \neg\mathsf{P}(x)}{\mathsf{Q}(x) \lor \mathsf{R}(x,y)} \quad \neg\mathsf{Q}(\mathsf{a})}{\mathsf{R}(\mathsf{a},y)} \; \sigma = \{x \mapsto \mathsf{a}\}$$

$$\cfrac{\overset{\sqcap}{\mathsf{R}(\mathsf{a},y)} \quad \cfrac{\cfrac{\cfrac{\mathsf{S}(\mathsf{a},y) \lor \neg\mathsf{R}(\mathsf{a},y) \lor \mathsf{S}(x,\mathsf{b})}{\mathsf{S}(\mathsf{a},\mathsf{b}) \lor \neg\mathsf{R}(\mathsf{a},\mathsf{b})}\,\sigma_1 \quad \neg\mathsf{S}(\mathsf{a},\mathsf{b}) \lor \neg\mathsf{R}(\mathsf{a},\mathsf{b})}{\neg\mathsf{R}(\mathsf{a},\mathsf{b}) \lor \neg\mathsf{R}(\mathsf{a},\mathsf{b})}}{\neg\mathsf{R}(\mathsf{a},\mathsf{b})}}{\square}\,\sigma_2$$

Intermission

Summary Last Lecture

Definition

- a literal L is maximal if $\exists$ ground σ such that for no other literal M: $M\sigma \succ_{\mathsf{L}} L\sigma$
- L is strictly maximal if $\exists$ ground σ such that for no other literal M: $M\sigma \succcurlyeq_{\mathsf{L}} L\sigma$; here $\succcurlyeq_{\mathsf{L}}$ denotes the reflexive closure

Definition

ordered resolution

$$\frac{C \vee A \quad D \vee \neg B}{(C \vee D)\sigma}$$

ordered factoring

$$\frac{C \vee A \vee B}{(C \vee A)\sigma}$$

1. σ is a mgu of the atomic formulas A and B
2. $A\sigma$ is strictly maximal with respect to $C\sigma$; $\neg B\sigma$ is maximal with respect to $D\sigma$

Outline of the Lecture

Early Approaches in Automated Reasoning

Herbrand's theorem for dummies, Gilmore's prover, method of Davis and Putnam

Starting Points

resolution, tableau provers, Skolemisation, ordered resolution, redundancy and deletion

Automated Reasoning with Equality

paramodulation, ordered completion and proof orders, superposition

Applications of Automated Reasoning

Neuman-Stubblebinde Key Exchange Protocol, Robbins problem

Definition

- define the ordered resolution operator $\mathsf{Res}_{\mathsf{OR}}(\mathcal{C})$ as follows:

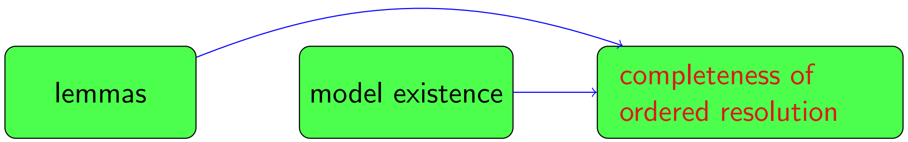
$$\mathsf{Res}_{\mathsf{OR}}(\mathcal{C}) = \{D \mid D \text{ is ordered res./factor with premises in } \mathcal{C}\}$$

- n^{th} (unrestricted) iteration $\mathsf{Res}^n_{\mathsf{OR}}$ ($\mathsf{Res}^*_{\mathsf{OR}}$) of the operator $\mathsf{Res}_{\mathsf{OR}}$ is defined as for unrestricted resolution

Theorem

ordered resolution is sound and complete; let F be a sentence and $\mathcal{C}$ its clause form; then F is unsatisfiable iff $\square \in \mathsf{Res}^*_{\mathsf{OR}}(\mathcal{C})$

Proof Plan.



■

Recall

- let $\mathcal{G}$ be a set of universal sentences (of $\mathcal{L}$) without $=$
- $\mathcal{G}$ has a Herbrand model or $\mathcal{G}$ is unsatisfiable; in the latter case the following statements hold (and are equivalent):
  1. $\exists$ finite subset $S \subseteq \mathsf{Gr}(\mathcal{G})$; conjunction $\bigwedge S$ is unsatisfiable
  2. $\exists$ finite subset $S \subseteq \mathsf{Gr}(\mathcal{G})$; disjunction $\bigvee\{\neg A \mid A \in S\}$ is valid

Proof of Completeness.

1. extend $\succ_{\mathsf{L}}$ to an order on clauses $\succ_{\mathsf{C}}$
2. a clause set $\mathcal{C}$ is maximal if

$$\neg\exists\mathcal{D} = \mathcal{D}' \cup \{D\} \ \big(\mathcal{C} = \mathcal{D}' \cup \{D_1, \dots, D_n\}, \forall i\ D \succ_{\mathsf{C}} D_i \text{ and there is no } E \in \mathcal{D}',\ E \succ_{\mathsf{C}} D\big)$$

3. choose a maximal unsatisfiable clause set $\mathcal{C}$
   continue according to proof plan

this proves ground completeness; completeness follows by reformulation of the lifting lemmas ■

Lock Resolution

Definition

a pair (L, i), L a literal, $i \in \mathbb{N}$ is an indexed literal; different literals are indexed with different numbers

Definition

lock resolution

$$\frac{C \vee (A, i) \quad D \vee (\neg B, j)}{(C \vee D)\sigma}$$

lock factoring

$$\frac{C \vee (A, i) \vee (B, k)}{(C \vee (A, i))\sigma}$$

1. σ is a mgu of the atomic formulas A and B
2. i is minimal with respect to C; j is minimal with respect to D
3. i is minimal with respect to $C \vee (B, k)$, $i \leqslant k$

Remark

indexing represents an a priori literal order, blind on substitutions

Example

consider the indexed clause set $\mathcal{C} = \{\neg \overset{1}{\mathsf{P}}(x), \neg \overset{3}{\mathsf{Q}}(\mathsf{a}), \neg \overset{5}{\mathsf{S}}(\mathsf{a}, \mathsf{b}) \vee \neg \overset{8}{\mathsf{R}}(\mathsf{a}, \mathsf{b}), \overset{2}{\mathsf{P}}(x) \vee \overset{4}{\mathsf{Q}}(x) \vee \overset{10}{\mathsf{R}}(x, y), \overset{6}{\mathsf{S}}(\mathsf{a}, y) \vee \neg \overset{9}{\mathsf{R}}(\mathsf{a}, y) \vee \overset{7}{\mathsf{S}}(x, \mathsf{b})\}$

$$\Pi \qquad \frac{\dfrac{\overset{2}{\mathsf{P}}(x) \vee \overset{4}{\mathsf{Q}}(x) \vee \overset{10}{\mathsf{R}}(x, y) \quad \neg \overset{1}{\mathsf{P}}(x)}{\overset{4}{\mathsf{Q}}(x) \vee \overset{10}{\mathsf{R}}(x, y)} \qquad \neg \overset{3}{\mathsf{Q}}(\mathsf{a})}{\overset{10}{\mathsf{R}}(\mathsf{a}, y)}\ \sigma = \{x \mapsto \mathsf{a}\}$$

$$\frac{\overset{\Pi}{\overset{10}{\mathsf{R}}(\mathsf{a}, y)} \qquad \dfrac{\dfrac{\dfrac{\overset{6}{\mathsf{S}}(\mathsf{a}, y) \vee \neg \overset{9}{\mathsf{R}}(\mathsf{a}, y) \vee \overset{7}{\mathsf{S}}(x, \mathsf{b})}{\overset{6}{\mathsf{S}}(\mathsf{a}, \mathsf{b}) \vee \neg \overset{9}{\mathsf{R}}(\mathsf{a}, \mathsf{b})}\ \sigma_1 \qquad \neg \overset{5}{\mathsf{S}}(\mathsf{a}, \mathsf{b}) \vee \neg \overset{8}{\mathsf{R}}(\mathsf{a}, \mathsf{b})}{\neg \overset{8}{\mathsf{R}}(\mathsf{a}, \mathsf{b}) \vee \neg \overset{9}{\mathsf{R}}(\mathsf{a}, \mathsf{b})}}{\neg \overset{8}{\mathsf{R}}(\mathsf{a}, \mathsf{b})}}{\square}\ \sigma_2$$

Definition

- define the lock resolution operator $\mathsf{Res}_{\mathsf{L}}(\mathcal{C})$ as follows:

  $$\mathsf{Res}_{\mathsf{L}}(\mathcal{C}) = \{D \mid D \text{ is lock res./factor with premises in } \mathcal{C}\}$$

- n^{th} (unrestricted) iteration $\mathsf{Res}_{\mathsf{L}}^{n}$ ($\mathsf{Res}_{\mathsf{L}}^{*}$) of the operator $\mathsf{Res}_{\mathsf{L}}$ is defined as for unrestricted resolution

Theorem

lock resolution is sound and complete; let F be a sentence and $\mathcal{C}$ its clause form; then F is unsatisfiable iff $\square \in \mathsf{Res}_{\mathsf{L}}^{*}(\mathcal{C})$

Proof.

lock resolution is a refinement, thus soundness is trivial; completeness follows as for ordered resolution ■

Redundancy and Deletion

Definition

define resolution operator $\mathsf{Res}(\mathcal{C})$

- $\mathsf{Res}(\mathcal{C}) = \{D \mid D \text{ is resolvent or factor with premises in } \mathcal{C}\}$
- $\mathsf{Res}^0(\mathcal{C}) = \mathcal{C}$; $\mathsf{Res}^{n+1}(\mathcal{C}) := \mathsf{Res}^n(\mathcal{C}) \cup \mathsf{Res}(\mathsf{Res}^n(\mathcal{C}))$
- $\mathsf{Res}^*(\mathcal{C}) := \bigcup_{n \geqslant 0} \mathsf{Res}^n(\mathcal{C})$

Definition

- let $\mathsf{d}(\mathcal{C}) = \min\{n \mid \square \in \mathsf{Res}^n(\mathcal{C})\}$
- the search complexity of Res wrt clause set $\mathcal{C}$ is $\mathsf{scomp}(\mathcal{C}) = |\mathsf{Res}^{\mathsf{d}(\mathcal{C})}(\mathcal{C})|$

Question

howto reduce the search complexity (of resolution refinements)?

Answer

three answers:

1. refinements
   consider refutational complete restrictions of resolution
2. redundancy tests
   redundancy can appear in the form of circular derivations or in that of tautology clauses
3. heuristics
   ...

Remarks

- refinements reduce the search space as fewer derivations are possible, however the minimal proof length may be increased
- redundancy tests cannot increase the proof length, but may be costly

call a clause D redundant in $\mathcal{C}$ if $\exists C_1, \dots, C_k$ with $C_1, \dots, C_k \models D$

Lemma

application of subsumption and tautology elimination as pre-procession steps preserves completeness

Definition

subsumption and resolution can be combined in the following ways

1. forward subsumption
   newly derived clauses subsumed by existing clauses are deleted
2. backward subsumption
   existing clauses C subsumed by newly derived clauses D become inactive; inactive clauses have to be reactivated, if D is no longer an ancestor of current clause (e.g. D has been deleted)
3. replacement
   the set of all clauses (derived and initial) are frequently reduced under subsumption

Tautology Elimination

Definition

- a clause C containing complementary literals is a tautology
- tautology elimination is the process of removing newly derived tautological clauses (that is, we assume the initial clause set is taut-reduced)

Example

consider the clause

$$\mathsf{P}(\mathsf{f}(\mathsf{a},\mathsf{b})) \lor \neg \mathsf{P}(\mathsf{f}(x,\mathsf{b})) \lor \neg \mathsf{P}(\mathsf{f}(\mathsf{a},y))$$

factoring yields the tautology $\mathsf{P}(\mathsf{f}(\mathsf{a},\mathsf{b})) \lor \neg \mathsf{P}(\mathsf{f}(\mathsf{a},\mathsf{b}))$

Example

consider the following (tautology free) clause set $\mathcal{C}$

$$\mathsf{P}(x) \lor \mathsf{R}(x) \quad \mathsf{R}(x) \lor \neg\mathsf{P}(x) \quad \mathsf{P}(x) \lor \neg\mathsf{R}(x) \quad \neg\mathsf{P}(x) \lor \neg\mathsf{R}(x)$$

we have $\text{scomp}(\mathcal{C}) = 15$ for unrestricted resolution; however the following resolution steps derive tautologies

$$\frac{\mathsf{P}(x) \lor \mathsf{R}(x) \quad \neg\mathsf{P}(x) \lor \neg\mathsf{R}(x)}{\mathsf{P}(x) \lor \neg\mathsf{P}(x)} \qquad \frac{\mathsf{P}(x) \lor \mathsf{R}(x) \quad \neg\mathsf{P}(x) \lor \neg\mathsf{R}(x)}{\mathsf{R}(x) \lor \neg\mathsf{R}(x)}$$

Lemma

1. *tautology elimination is not complete for lock resolution*
2. *tautology elimination is complete for unrestricted and ordered resolution*

Theorem

1. *(ordered) resolution (for any well order $\succ$ on ground atoms) is complete under forward subsumption*
2. *forward subsumption does not increase the search complexity of (ordered) resolution*

Proof Sketch.

1. let C', C, D', D be clauses such that C' subsumes C and D' subsumes D
2. one shows that if E is a resolvent of C and D, then one of the following cases happens:
   - C' subsumes E
   - D' subsumes E
   - $\exists$ resolvent E' of C' and D' such that E' subsumes E
3. using this observation in an inductive argument, completeness follows

■

Lemma

lock resolution is not complete under forward subsumption

Proof.

1. let C, D be indexed clauses; we say an C subsumes D if the clause part of C subsumes the clause part of D
2. consider the following clause set $\mathcal{C}$

$$\overset{5}{\mathrm{P}}(x) \lor \overset{1}{\mathrm{R}}(x) \quad \overset{6}{\mathrm{R}}(x) \lor \neg\overset{3}{\mathrm{P}}(x) \quad \overset{4}{\mathrm{P}}(x) \lor \neg\overset{7}{\mathrm{R}}(x) \quad \neg\overset{8}{\mathrm{P}}(x) \lor \neg\overset{2}{\mathrm{R}}(x)$$

3. the following clauses are derivable by lock resolution and essential to derive $\square$

$$\overset{6}{\mathrm{R}}(x) \lor \neg\overset{8}{\mathrm{P}}(x) \qquad \neg\overset{8}{\mathrm{P}}(x) \lor \neg\overset{7}{\mathrm{R}}(x)$$

4. however these are subsumed by $\overset{6}{\mathrm{R}}(x) \lor \neg\overset{3}{\mathrm{P}}(x)$ and $\neg\overset{8}{\mathrm{P}}(x) \lor \neg\overset{2}{\mathrm{R}}(x)$ respectively

■

Example

consider the following set of clauses

$$C_1\colon\ \mathrm{P}(\mathrm{f}(x)) \lor \mathrm{R}(x) \lor \neg\mathrm{P}(\mathrm{f}(x)) \qquad C_2\colon\ \mathrm{P}(x) \lor \mathrm{Q}(x)$$
$$C_3\colon\ \mathrm{R}(\mathrm{f}(x)) \qquad C_4\colon\ \mathrm{Q}(x) \lor \neg\mathrm{R}(x)$$
$$C_5\colon\ \neg\mathrm{Q}(\mathrm{f}(x))$$

C_1 can be resolved with C_2, C_4 and itself

Lemma

let C and D be clauses and C a tautology; any resolvent of C and D is either a tautology or subsumed by D

Theorem

(ordered) resolution is complete under forward subsumption and tautology elimination