Automated Reasoning

Georg Moser

Institute of Computer Science @ UIBK

Winter 2013

Summary of Last Lecture

Ordered Completion

deduction	$\mathcal{E};\mathcal{R} \vdash \mathcal{E} \cup \{s = t\};\mathcal{R}$	
	if $s \leftrightarrow_{\mathcal{E}\cup\mathcal{R}} w \leftrightarrow_{\mathcal{E}\cup\mathcal{R}} t$, $s \not\preceq w$, $t \not\preceq w$	
orientation	$\mathcal{E} \cup \{s = t\};\mathcal{R} \vdash \mathcal{E};\mathcal{R} \cup \{s \to t\}$	if $s \succ t$
deletion	$\mathcal{E} \cup \{s = s\};\mathcal{R} \vdash \mathcal{E};\mathcal{R}$	
simplification	$\mathcal{E} \cup \{s = t\};\mathcal{R} \vdash \mathcal{E} \cup \{u = t\};\mathcal{R}$	if $s \to_{\mathcal{R}} u$
composition	$\mathcal{E};\mathcal{R} \cup \{s \to t\} \vdash \mathcal{E};\mathcal{R} \cup \{s \to u\}$	if $t \to_{\mathcal{R}} u$
collapse	$\mathcal{E};\mathcal{R} \cup \{s[w] \to t\} \vdash \mathcal{E} \cup \{s[u] = t\};\mathcal{R}$	
	if $w \to_{\mathcal{R}} u$ and either $t \succ u$ or $w \neq s[w]$	

Definition

- a proof of $s = t$ wrt $\mathcal{E};\mathcal{R}$ is ...
- a proof of form ... is called rewrite proof

Outline of the Lecture

Early Approaches in Automated Reasoning

short recollection of Herbrand's theorem, 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, group theory, resolution and paramodulation as decision procedure, ...

Outline of the Lecture

Early Approaches in Automated Reasoning

short recollection of Herbrand's theorem, 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, group theory, resolution and paramodulation as decision procedure, . . .

Definition

s encompasses t if $s = C[t\sigma]$ for some context C and some substitution σ

Definition

s encompasses t if $s = C[t\sigma]$ for some context C and some substitution σ

Definition

cost measure of proof steps

$$\text{cost of } s[u] \ \rho \ s[v] = \begin{cases} (\{s[u]\}, u, \rho, s[v]) & \text{if } s[u] \succ s[v] \\ (\{s[v]\}, v, \rho, s[u]) & \text{if } s[v] \succ s[u] \\ (\{s[u], s[v]\}, \bot, \bot, \bot) & \text{otherwise} \end{cases}$$

cost measure is lexicographically compared as follows:

Definition

s encompasses t if $s = C[t\sigma]$ for some context C and some substitution σ

Definition

cost measure of proof steps

$$\text{cost of } s[u] \;\rho\; s[v] = \begin{cases} (\{s[u]\}, u, \rho, s[v]) & \text{if } s[u] \succ s[v] \\ (\{s[v]\}, v, \rho, s[u]) & \text{if } s[v] \succ s[u] \\ (\{s[u], s[v]\}, \bot, \bot, \bot) & \text{otherwise} \end{cases}$$

cost measure is lexicographically compared as follows:

1. multiset extension of $\succ$

Definition

s encompasses t if $s = C[t\sigma]$ for some context C and some substitution σ

Definition

cost measure of proof steps

$$\text{cost of } s[u] \;\rho\; s[v] = \begin{cases} (\{s[u]\}, u, \rho, s[v]) & \text{if } s[u] \succ s[v] \\ (\{s[v]\}, v, \rho, s[u]) & \text{if } s[v] \succ s[u] \\ (\{s[u], s[v]\}, \bot, \bot, \bot) & \text{otherwise} \end{cases}$$

cost measure is lexicographically compared as follows:

1. multiset extension of $\succ$
2. encompassment order

Definition

s encompasses t if $s = C[t\sigma]$ for some context C and some substitution σ

Definition

cost measure of proof steps

$$\text{cost of } s[u] \;\rho\; s[v] = \begin{cases} (\{s[u]\}, u, \rho, s[v]) & \text{if } s[u] \succ s[v] \\ (\{s[v]\}, v, \rho, s[u]) & \text{if } s[v] \succ s[u] \\ (\{s[u], s[v]\}, \bot, \bot, \bot) & \text{otherwise} \end{cases}$$

cost measure is lexicographically compared as follows:

1. multiset extension of $\succ$
2. encompassment order
3. some order with $\leftrightarrow \; > \; \rightarrow$ and $\leftrightarrow \; > \; \leftarrow$

Definition

s encompasses t if $s = C[t\sigma]$ for some context C and some substitution σ

Definition

cost measure of proof steps

$$\text{cost of } s[u] \; \rho \; s[v] = \begin{cases} (\{s[u]\}, u, \rho, s[v]) & \text{if } s[u] \succ s[v] \\ (\{s[v]\}, v, \rho, s[u]) & \text{if } s[v] \succ s[u] \\ (\{s[u], s[v]\}, \bot, \bot, \bot) & \text{otherwise} \end{cases}$$

cost measure is lexicographically compared as follows:

1. multiset extension of $\succ$
2. encompassment order
3. some order with $\leftrightarrow \; > \; \rightarrow$ and $\leftrightarrow \; > \; \leftarrow$
4. reduction order $\succ$

Definition

s encompasses t if $s = C[t\sigma]$ for some context C and some substitution σ

Definition

cost measure of proof steps

$$\text{cost of } s[u] \ \rho \ s[v] = \begin{cases} (\{s[u]\}, u, \rho, s[v]) & \text{if } s[u] \succ s[v] \\ (\{s[v]\}, v, \rho, s[u]) & \text{if } s[v] \succ s[u] \\ (\{s[u], s[v]\}, \bot, \bot, \bot) & \text{otherwise} \end{cases}$$

cost measure is lexicographically compared as follows:

1. multiset extension of $\succ$
2. encompassment order
3. some order with $\leftrightarrow > \rightarrow$ and $\leftrightarrow > \leftarrow$
4. reduction order $\succ$

$\bot$ is supposed to be minimal in all orders;

Definition

s encompasses t if $s = C[t\sigma]$ for some context C and some substitution σ

Definition

cost measure of proof steps

$$\text{cost of } s[u] \ \rho \ s[v] = \begin{cases} (\{s[u]\}, u, \rho, s[v]) & \text{if } s[u] \succ s[v] \\ (\{s[v]\}, v, \rho, s[u]) & \text{if } s[v] \succ s[u] \\ (\{s[u], s[v]\}, \bot, \bot, \bot) & \text{otherwise} \end{cases}$$

cost measure is lexicographically compared as follows:

1. multiset extension of $\succ$
2. encompassment order
3. some order with $\leftrightarrow > \rightarrow$ and $\leftrightarrow > \leftarrow$
4. reduction order $\succ$

$\bot$ is supposed to be minimal in all orders; let $\succ_\pi$ the multiset extension of the cost measure; then $\succ_\pi$ denotes a well-founded order on proofs

Fact

each completion step decreases the cost of certain proofs

Fact

each completion step decreases the cost of certain proofs

Proof Sketch.

Fact

each completion step decreases the cost of certain proofs

Proof Sketch.

- consider orientation that replaces an equation $s = t$ by rule $s \to t$

Fact

each completion step decreases the cost of certain proofs

Proof Sketch.

- consider orientation that replaces an equation $s = t$ by rule $s \to t$
- yields proof transformation

$$(u[s\sigma] \leftrightarrow u[t\sigma]) \Rightarrow (u[s\sigma] \to u[t\sigma])$$

Fact

each completion step decreases the cost of certain proofs

Proof Sketch.

- consider orientation that replaces an equation $s = t$ by rule $s \to t$
- yields proof transformation

$$(u[s\sigma] \leftrightarrow u[t\sigma]) \Rightarrow (u[s\sigma] \to u[t\sigma])$$

- cost of $(u[s\sigma] \leftrightarrow u[t\sigma]) >$ cost of $(u[s\sigma] \to u[t\sigma])$

Fact

each completion step decreases the cost of certain proofs

Proof Sketch.

- consider orientation that replaces an equation $s = t$ by rule $s \to t$
- yields proof transformation

$$(u[s\sigma] \leftrightarrow u[t\sigma]) \Rightarrow (u[s\sigma] \to u[t\sigma])$$

- cost of $(u[s\sigma] \leftrightarrow u[t\sigma]) >$ cost of $(u[s\sigma] \to u[t\sigma])$

$\blacksquare$

Fact

each completion step decreases the cost of certain proofs

Proof Sketch.

- consider orientation that replaces an equation $s = t$ by rule $s \to t$
- yields proof transformation

$$(u[s\sigma] \leftrightarrow u[t\sigma]) \Rightarrow (u[s\sigma] \to u[t\sigma])$$

- cost of $(u[s\sigma] \leftrightarrow u[t\sigma])$ > cost of $(u[s\sigma] \to u[t\sigma])$

∎

recall: $\mathcal{E}_\infty = \bigcup_{i \geqslant 0} \bigcap_{j \geqslant i} \mathcal{E}_j$; $\mathcal{R}_\infty = \bigcup_{i \geqslant 0} \bigcap_{j \geqslant i} \mathcal{R}_j$

Fact

each completion step decreases the cost of certain proofs

Proof Sketch.

- consider orientation that replaces an equation $s = t$ by rule $s \to t$
- yields proof transformation

$$(u[s\sigma] \leftrightarrow u[t\sigma]) \Rightarrow (u[s\sigma] \to u[t\sigma])$$

- cost of $(u[s\sigma] \leftrightarrow u[t\sigma])$ > cost of $(u[s\sigma] \to u[t\sigma])$

∎

recall: $\mathcal{E}_\infty = \bigcup_{i \geqslant 0} \bigcap_{j \geqslant i} \mathcal{E}_j$; $\mathcal{R}_\infty = \bigcup_{i \geqslant 0} \bigcap_{j \geqslant i} \mathcal{R}_j$

Definition

a derivation is fair if each ordered critical pair $u = v \in \mathcal{E}_\infty \cup \mathcal{R}_\infty$ is an element of some $\mathcal{E}_i$

Theorem

let $(\mathcal{E}_0; \mathcal{R}_0), (\mathcal{E}_1; \mathcal{R}_1), \ldots$ *be a fair ordered completion derivation with* $\mathcal{R}_0 = \varnothing$*; then the following is equivalent:*

1. $s = t$ *is a consequence of* $\mathcal{E}_0$

Theorem

let $(\mathcal{E}_0; \mathcal{R}_0), (\mathcal{E}_1; \mathcal{R}_1), \dots$ *be a fair ordered completion derivation with* $\mathcal{R}_0 = \varnothing$*; then the following is equivalent:*

1. $s = t$ *is a consequence of* $\mathcal{E}_0$
2. $s = t$ *has a rewrite proof in* $\mathcal{E}_\infty^{\succ} \cup \mathcal{R}_\infty$

Theorem

let $(\mathcal{E}_0; \mathcal{R}_0), (\mathcal{E}_1; \mathcal{R}_1), \dots$ *be a fair ordered completion derivation with* $\mathcal{R}_0 = \varnothing$*; then the following is equivalent:*

1. $s = t$ *is a consequence of* $\mathcal{E}_0$
2. $s = t$ *has a rewrite proof in* $\mathcal{E}_\infty^\succ \cup \mathcal{R}_\infty$
3. $\exists$ i *such that* $s = t$ *has a rewrite proof in* $\mathcal{E}_i^\succ \cup \mathcal{R}_i$

Theorem

let $(\mathcal{E}_0; \mathcal{R}_0), (\mathcal{E}_1; \mathcal{R}_1), \dots$ *be a fair ordered completion derivation with* $\mathcal{R}_0 = \varnothing$*; then the following is equivalent:*

1. $s = t$ *is a consequence of* $\mathcal{E}_0$
2. $s = t$ *has a rewrite proof in* $\mathcal{E}_\infty^{\succ} \cup \mathcal{R}_\infty$
3. $\exists\ i$ *such that* $s = t$ *has a rewrite proof in* $\mathcal{E}_i^{\succ} \cup \mathcal{R}_i$

Definitions

- let $\mathcal{E}$ be a set of equations and $s = t$ an equation (possibly containing variables); then $\mathcal{E} \models s = t$ is the word problem for $\mathcal{E}$
- the word problem becomes a refutation theorem proving problem once we consider the clause form of the negation of the word problem:
  1. a set of positive unit equations in $\mathcal{E}$
  2. a ground disequation obtained by negation and Skolemisation of $s = t$

Completeness of Superposition

Corollary

superposition with equations is sound and complete, that is, if $\mathcal{C}$ *is the clause representation of the (negated) word problem* $\mathcal{E} \models s = t$*, then the saturation of* $\mathcal{C}$ *wrt to superposition (and equality resolution) contains* $\square$ *iff* $\mathcal{E} \models s = t$

Completeness of Superposition

Corollary

superposition with equations is sound and complete, that is, if $\mathcal{C}$ is the clause representation of the (negated) word problem $\mathcal{E} \models s = t$, then the saturation of $\mathcal{C}$ wrt to superposition (and equality resolution) contains $\square$ iff $\mathcal{E} \models s = t$

Proof.

1. let $\mathcal{C}'$ denote the saturation and let $\square \in \mathcal{C}'$

Completeness of Superposition

Corollary

superposition with equations is sound and complete, that is, if $\mathcal{C}$ is the clause representation of the (negated) word problem $\mathcal{E} \models s = t$, then the saturation of $\mathcal{C}$ wrt to superposition (and equality resolution) contains $\Box$ iff $\mathcal{E} \models s = t$

Proof.

1. let $\mathcal{C}'$ denote the saturation and let $\Box \in \mathcal{C}'$
2. then $\mathcal{E} \models s = t$ due to soundness of superposition

Completeness of Superposition

Corollary

superposition with equations is sound and complete, that is, if $\mathcal{C}$ is the clause representation of the (negated) word problem $\mathcal{E} \models s = t$, then the saturation of $\mathcal{C}$ wrt to superposition (and equality resolution) contains $\square$ iff $\mathcal{E} \models s = t$

Proof.

1. let $\mathcal{C}'$ denote the saturation and let $\square \in \mathcal{C}'$
2. then $\mathcal{E} \models s = t$ due to soundness of superposition
3. otherwise assume $\square \notin \mathcal{C}'$

Completeness of Superposition

Corollary

superposition with equations is sound and complete, that is, if $\mathcal{C}$ is the clause representation of the (negated) word problem $\mathcal{E} \models s = t$, then the saturation of $\mathcal{C}$ wrt to superposition (and equality resolution) contains $\square$ iff $\mathcal{E} \models s = t$

Proof.

1. let $\mathcal{C}'$ denote the saturation and let $\square \in \mathcal{C}'$
2. then $\mathcal{E} \models s = t$ due to soundness of superposition
3. otherwise assume $\square \notin \mathcal{C}'$
4. then $s = t$ does not have a proof in $\mathcal{C}'$

Completeness of Superposition

Corollary

superposition with equations is sound and complete, that is, if $\mathcal{C}$ is the clause representation of the (negated) word problem $\mathcal{E} \models s = t$, then the saturation of $\mathcal{C}$ wrt to superposition (and equality resolution) contains $\Box$ iff $\mathcal{E} \models s = t$

Proof.

1. let $\mathcal{C}'$ denote the saturation and let $\Box \in \mathcal{C}'$
2. then $\mathcal{E} \models s = t$ due to soundness of superposition
3. otherwise assume $\Box \notin \mathcal{C}'$
4. then $s = t$ does not have a proof in $\mathcal{C}'$
5. with the theorem we conclude that $\mathcal{E} \not\models s = t$

Completeness of Superposition

Corollary

superposition with equations is sound and complete, that is, if $\mathcal{C}$ is the clause representation of the (negated) word problem $\mathcal{E} \models s = t$, then the saturation of $\mathcal{C}$ wrt to superposition (and equality resolution) contains $\square$ iff $\mathcal{E} \models s = t$

Proof.

1. let $\mathcal{C}'$ denote the saturation and let $\square \in \mathcal{C}'$
2. then $\mathcal{E} \models s = t$ due to soundness of superposition
3. otherwise assume $\square \notin \mathcal{C}'$
4. then $s = t$ does not have a proof in $\mathcal{C}'$
5. with the theorem we conclude that $\mathcal{E} \not\models s = t$

∎

Superposition for Horn Clauses

Idea

- consider a set P of non-equational Horn clauses (= a logic program)
- define the operator:

$$T_P \colon I \mapsto \{A \mid A \leftarrow B_1, \ldots, B_k \in \mathrm{Gr}(P) \text{ and } \forall\, i\ B_i \in I\}$$

- consider the least fixed point $\bigcup_{n \geqslant 0} T_p^n(\varnothing)$ of T_p
- then $\bigcup_{n \geqslant 0} T_p^n(\varnothing)$ denotes the unique minimal model of P

Superposition for Horn Clauses

Idea

- consider a set P of non-equational Horn clauses (= a logic program)
- define the operator:

$$T_P \colon I \mapsto \{A \mid A \leftarrow B_1, \dots, B_k \in \mathrm{Gr}(P) \text{ and } \forall\ i\ B_i \in I\}$$

- consider the least fixed point $\bigcup_{n \geqslant 0} T_p^n(\varnothing)$ of T_p
- then $\bigcup_{n \geqslant 0} T_p^n(\varnothing)$ denotes the unique minimal model of P

$A \leftarrow B_1, \dots, B_k$ produces A, if $\forall\ i\ B_i \in T_p^n(\varnothing)$ but $A \notin T_p^n(\varnothing)$

Superposition for Horn Clauses

Idea

- consider a set P of non-equational Horn clauses (= a logic program)
- define the operator:

$$T_P \colon I \mapsto \{A \mid A \leftarrow B_1, \dots, B_k \in \mathrm{Gr}(P) \text{ and } \forall\ i\ B_i \in I\}$$

- consider the least fixed point $\bigcup_{n \geqslant 0} T_p^n(\varnothing)$ of T_p
- then $\bigcup_{n \geqslant 0} T_p^n(\varnothing)$ denotes the unique minimal model of P

$A \leftarrow B_1, \dots, B_k$ produces A, if $\forall\ i\ B_i \in T_p^n(\varnothing)$ but $A \notin T_p^n(\varnothing)$

Definition

an equational Horn clause $C \equiv (u_1 = v_1, \dots, u_k = v_k \to s = t)$ is reductive for $s \to t$ (wrt to a reduction order $\succ$) if s is strictly maximal in C: (i) $s \succ t$, (ii) for all i: $s \succ u_i$, and (iii) for all i: $s \succ v_i$

NB: if C is reductive for $s \to t$, we write C as
$u_1 = v_1, \ldots, u_k = v_k \supset s \to t$

NB: if C is reductive for $s \to t$, we write C as
$u_1 = v_1, \ldots, u_k = v_k \supset s \to t$

Definition

- let $\mathcal{R}$ be a set of reductive clauses
- $\mathcal{R}$ induces the rewrite relation $\to_{\mathcal{R}}$: $s \to_{\mathcal{R}} t$ if
  1. $\exists$ reductive clause $C \supset l \to r$
  2. $\exists$ substitution σ such that $s = l\sigma$, $t = r\sigma$
  3. $\forall$ $u' = v' \in C$: $u'\sigma \downarrow v'\sigma$

NB: if C is reductive for $s \to t$, we write C as
$u_1 = v_1, \ldots, u_k = v_k \supset s \to t$

Definition

- let $\mathcal{R}$ be a set of reductive clauses
- $\mathcal{R}$ induces the rewrite relation $\to_{\mathcal{R}}$: $s \to_{\mathcal{R}} t$ if
  1. $\exists$ reductive clause $C \supset l \to r$
  2. $\exists$ substitution σ such that $s = l\sigma$, $t = r\sigma$
  3. $\forall\ u' = v' \in C$: $u'\sigma \downarrow v'\sigma$

Definition (superposition of reductive conditional rewrite rules)

$$\frac{C \supset s \to t \quad D \supset w[u] \to v}{(C, D \supset w[t] \to v)\sigma}$$

σ is mgu of s and u and u is not a variable

Definitions

- $(C, D \supset w[t] \to v)\sigma$ is a conditional critical pair
- $(C, D \supset w[t] \to v)\sigma$ converges if $\forall\ \tau$ such that $C\sigma\tau$ and $D\sigma\tau$ converge: $w[t]\sigma\tau \downarrow v\sigma\tau$

Definitions

- $(C, D \supset w[t] \to v)\sigma$ is a conditional critical pair
- $(C, D \supset w[t] \to v)\sigma$ converges if $\forall\ \tau$ such that $C\sigma\tau$ and $D\sigma\tau$ converge: $w[t]\sigma\tau \downarrow v\sigma\tau$

Lemma

a reductive conditional rewrite system is confluent iff all critical pairs converge

Definitions

- $(C, D \supset w[t] \to v)\sigma$ is a conditional critical pair
- $(C, D \supset w[t] \to v)\sigma$ converges if $\forall\ \tau$ such that $C\sigma\tau$ and $D\sigma\tau$ converge: $w[t]\sigma\tau \downarrow v\sigma\tau$

Lemma

a reductive conditional rewrite system is confluent iff all critical pairs converge

Theorem

let $\succ$ be a reduction order and let $\mathcal{C}$ be a set of reductive equational Horn clauses; then the word problem is decidable if all critical pairs converge

Superposition Calculus

Definition

$$\frac{C \vee A \quad D \vee \neg B}{(C \vee D)\sigma}\ \mathsf{ORe} \qquad \frac{C \vee A \vee B}{(C \vee A)\sigma}\ \mathsf{OFc}$$

$$\frac{C \vee s = t \quad D \vee \neg A[s']}{(C \vee D \vee \neg A[t])\sigma}\ \mathsf{OPm(L)} \qquad \frac{C \vee s = t \quad D \vee A[s']}{(C \vee D \vee A[t])\sigma}\ \mathsf{OPm(R)}$$

$$\frac{C \vee s = t \quad D \vee u[s'] \neq v}{(C \vee D \vee u[t] \neq v)\sigma}\ \mathsf{SpL} \qquad \frac{C \vee s = t \quad D \vee u[s'] = v}{(C \vee D \vee u[t] = v)\sigma}\ \mathsf{SpR}$$

$$\frac{C \vee s \neq t}{C\sigma}\ \mathsf{ERR} \qquad \frac{C \vee u = v \vee s = t}{(C \vee v \neq t \vee u = t)\sigma}\ \mathsf{EFc}$$

- ORe and OFc are ordered resolution and ordered factoring
- OPm(L), OPm(R), SpL, SpR stands for ordered paramodulation and superpostion (left or right)
- ERR means equality resolution and EFc means equality factoring

Definition (Definition (cont'd))

$$\frac{C \vee A \quad D \vee \neg B}{(C \vee D)\sigma}\ \mathsf{ORe} \qquad \frac{C \vee A \vee B}{(C \vee A)\sigma}\ \mathsf{OFc}$$

$$\frac{C \vee s = t \quad D \vee \neg A[s']}{(C \vee D \vee \neg A[t])\sigma}\ \mathsf{OPm(L)} \qquad \frac{C \vee s = t \quad D \vee A[s']}{(C \vee D \vee A[t])\sigma}\ \mathsf{OPm(R)}$$

$$\frac{C \vee s = t \quad D \vee u[s'] \neq v}{(C \vee D \vee u[t] \neq v)\sigma}\ \mathsf{SpL} \qquad \frac{C \vee s = t \quad D \vee u[s'] = v}{(C \vee D \vee u[t] = v)\sigma}\ \mathsf{SpR}$$

$$\frac{C \vee s \neq t}{C\sigma}\ \mathsf{ERR} \qquad \frac{C \vee u = v \vee s = t}{(C \vee v \neq t \vee u = t)\sigma}\ \mathsf{EFc}$$

Definition (Definition (cont'd))

$$\frac{C \vee A \quad D \vee \neg B}{(C \vee D)\sigma}\ \text{ORe} \qquad \frac{C \vee A \vee B}{(C \vee A)\sigma}\ \text{OFc}$$

$$\frac{C \vee s = t \quad D \vee \neg A[s']}{(C \vee D \vee \neg A[t])\sigma}\ \text{OPm(L)} \qquad \frac{C \vee s = t \quad D \vee A[s']}{(C \vee D \vee A[t])\sigma}\ \text{OPm(R)}$$

$$\frac{C \vee s = t \quad D \vee u[s'] \neq v}{(C \vee D \vee u[t] \neq v)\sigma}\ \text{SpL} \qquad \frac{C \vee s = t \quad D \vee u[s'] = v}{(C \vee D \vee u[t] = v)\sigma}\ \text{SpR}$$

$$\frac{C \vee s \neq t}{C\sigma}\ \text{ERR} \qquad \frac{C \vee u = v \vee s = t}{(C \vee v \neq t \vee u = t)\sigma}\ \text{EFc}$$

constraints:

Definition (Definition (cont'd))

$$\frac{C \vee A \quad D \vee \neg B}{(C \vee D)\sigma}\ \text{ORe} \qquad \frac{C \vee A \vee B}{(C \vee A)\sigma}\ \text{OFc}$$

$$\frac{C \vee s = t \quad D \vee \neg A[s']}{(C \vee D \vee \neg A[t])\sigma}\ \text{OPm(L)} \qquad \frac{C \vee s = t \quad D \vee A[s']}{(C \vee D \vee A[t])\sigma}\ \text{OPm(R)}$$

$$\frac{C \vee s = t \quad D \vee u[s'] \neq v}{(C \vee D \vee u[t] \neq v)\sigma}\ \text{SpL} \qquad \frac{C \vee s = t \quad D \vee u[s'] = v}{(C \vee D \vee u[t] = v)\sigma}\ \text{SpR}$$

$$\frac{C \vee s \neq t}{C\sigma}\ \text{ERR} \qquad \frac{C \vee u = v \vee s = t}{(C \vee v \neq t \vee u = t)\sigma}\ \text{EFc}$$

constraints:

1. for ordered resolution: $A\sigma$ is strictly maximal with respect to $C\sigma$ and $\neg B\sigma$ is maximal with respect to $D\sigma$
2. for ordered factoring: $A\sigma$ is strictly maximal wrt $C\sigma$.

Definition (Definition (cont'd))

$$\frac{C \vee A \quad D \vee \neg B}{(C \vee D)\sigma}\ \text{ORe} \qquad \frac{C \vee A \vee B}{(C \vee A)\sigma}\ \text{OFc}$$

$$\frac{C \vee s = t \quad D \vee \neg A[s']}{(C \vee D \vee \neg A[t])\sigma}\ \text{OPm(L)} \qquad \frac{C \vee s = t \quad D \vee A[s']}{(C \vee D \vee A[t])\sigma}\ \text{OPm(R)}$$

$$\frac{C \vee s = t \quad D \vee u[s'] \neq v}{(C \vee D \vee u[t] \neq v)\sigma}\ \text{SpL} \qquad \frac{C \vee s = t \quad D \vee u[s'] = v}{(C \vee D \vee u[t] = v)\sigma}\ \text{SpR}$$

$$\frac{C \vee s \neq t}{C\sigma}\ \text{ERR} \qquad \frac{C \vee u = v \vee s = t}{(C \vee v \neq t \vee u = t)\sigma}\ \text{EFc}$$

constraints:

1. for the superposition rules: σ is a mgu of s and s', s' not a variable, $t\sigma \not\succeq s\sigma$, $v\sigma \not\succeq u[s']\sigma$, $(s = t)\sigma$ is strictly maximal wrt $C\sigma$
2. $\neg A[s']$ and $u[s'] \neq v$ are maximal, while $A[s']$ and $u[s'] = v$ are strictly maximal wrt $D\sigma$
3. $(s = t)\sigma \not\succeq (u[s'] = v)\sigma$

Definition (Definition (cont'd))

$$\frac{C \vee A \quad D \vee \neg B}{(C \vee D)\sigma}\ \mathsf{ORe} \qquad \frac{C \vee A \vee B}{(C \vee A)\sigma}\ \mathsf{OFc}$$

$$\frac{C \vee s = t \quad D \vee \neg A[s']}{(C \vee D \vee \neg A[t])\sigma}\ \mathsf{OPm(L)} \qquad \frac{C \vee s = t \quad D \vee A[s']}{(C \vee D \vee A[t])\sigma}\ \mathsf{OPm(R)}$$

$$\frac{C \vee s = t \quad D \vee u[s'] \neq v}{(C \vee D \vee u[t] \neq v)\sigma}\ \mathsf{SpL} \qquad \frac{C \vee s = t \quad D \vee u[s'] = v}{(C \vee D \vee u[t] = v)\sigma}\ \mathsf{SpR}$$

$$\frac{C \vee s \neq t}{C\sigma}\ \mathsf{ERR} \qquad \frac{C \vee u = v \vee s = t}{(C \vee v \neq t \vee u = t)\sigma}\ \mathsf{EFc}$$

constraints:

1. for the equality resolution rule: σ is a mgu of s and t
2. $(s \neq t)\sigma$ is maximal wrt $C\sigma$

Definition (Definition (cont'd))

$$\frac{C \vee A \quad D \vee \neg B}{(C \vee D)\sigma}\ \text{ORe} \qquad \frac{C \vee A \vee B}{(C \vee A)\sigma}\ \text{OFc}$$

$$\frac{C \vee s = t \quad D \vee \neg A[s']}{(C \vee D \vee \neg A[t])\sigma}\ \text{OPm(L)} \qquad \frac{C \vee s = t \quad D \vee A[s']}{(C \vee D \vee A[t])\sigma}\ \text{OPm(R)}$$

$$\frac{C \vee s = t \quad D \vee u[s'] \neq v}{(C \vee D \vee u[t] \neq v)\sigma}\ \text{SpL} \qquad \frac{C \vee s = t \quad D \vee u[s'] = v}{(C \vee D \vee u[t] = v)\sigma}\ \text{SpR}$$

$$\frac{C \vee s \neq t}{C\sigma}\ \text{ERR} \qquad \frac{C \vee u = v \vee s = t}{(C \vee v \neq t \vee u = t)\sigma}\ \text{EFc}$$

constraints:

1. for equality factoring: σ is mgu of s and u, $(s = t)\sigma$ is strictly maximal in $C\sigma$
2. $(s = t)\sigma \not\preccurlyeq (u = v)\sigma$

Definition

- define the superposition operator $\mathsf{Res}_{\mathsf{SP}}(\mathcal{C})$ as follows:

  $$\mathsf{Res}_{\mathsf{SP}}(\mathcal{C}) = \{D \mid D \text{ is conclusion of ORe–EFc with premises in } \mathcal{C}\}$$

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

Definition

- define the superposition operator $\mathrm{Res}_{\mathrm{SP}}(\mathcal{C})$ as follows:

  $$\mathrm{Res}_{\mathrm{SP}}(\mathcal{C}) = \{D \mid D \text{ is conclusion of ORe–EFc with premises in } \mathcal{C}\}$$

- n^{th} (unrestricted) iteration $\mathrm{Res}^{n}_{\mathrm{SP}}$ ($\mathrm{Res}^{*}_{\mathrm{SP}}$) of the operator $\mathrm{Res}_{\mathrm{SP}}$ is defined as above

Example

re-consider $\mathcal{C} = \{\mathsf{c} \neq \mathsf{d}, \mathsf{b} = \mathsf{d}, \mathsf{a} \neq \mathsf{d} \vee \mathsf{a} = \mathsf{c}, \mathsf{a} = \mathsf{b} \vee \mathsf{a} = \mathsf{d}\}$ together with the term order: $\mathsf{a} \succ \mathsf{b} \succ \mathsf{c} \succ \mathsf{d}$; without equality factoring only the following clause is derivable:

$$\mathsf{a} \neq \mathsf{d} \vee \mathsf{b} = \mathsf{c} \vee \mathsf{a} = \mathsf{d}$$

here the atom order is the multiset extension of $\succ$: $\mathsf{a} = \mathsf{b} \equiv \{\mathsf{a}, \mathsf{b}\} \succ \{\mathsf{a}, \mathsf{d}\} \equiv \mathsf{a} = \mathsf{d}$ and the literal order $\succ_{\mathsf{L}}$ is the multiset extenion of the atom order: $\mathsf{a} = \mathsf{c} \succ_{\mathsf{L}} \mathsf{a} \neq \mathsf{d}$

Candidate Models

Definitions

- let $\mathcal{O}$ be a clause inference operator

Candidate Models

Definitions

- let $\mathcal{O}$ be a clause inference operator
- let $\mathcal{I}$ denote a mapping that assigns to each ground clause set $\mathcal{C}$ an equality Herbrand interpretation, the candidate model $\mathcal{I}_{\mathcal{C}}$

Candidate Models

Definitions

- let $\mathcal{O}$ be a clause inference operator
- let $\mathcal{I}$ denote a mapping that assigns to each ground clause set $\mathcal{C}$ an equality Herbrand interpretation, the candidate model $\mathcal{I}_{\mathcal{C}}$
- if $\mathcal{I}_{\mathcal{C}} \not\models \mathcal{C}$ there $\exists$ minimal counter-example C

Candidate Models

Definitions

- let $\mathcal{O}$ be a clause inference operator
- let $\mathcal{I}$ denote a mapping that assigns to each ground clause set $\mathcal{C}$ an equality Herbrand interpretation, the candidate model $\mathcal{I}_{\mathcal{C}}$
- if $\mathcal{I}_{\mathcal{C}} \not\models \mathcal{C}$ there $\exists$ minimal counter-example C
- $\mathcal{O}$ has reduction property if
  1. $\forall\ \mathcal{C}$
  2. $\forall$ minimal counter-examples C for $\mathcal{I}_{\mathcal{C}}$
  3. $\exists$ inference from $\mathcal{C}$ in $\mathcal{O}$

$$\frac{C_1 \quad \ldots \quad C_n \quad C}{D}$$

where $\mathcal{I}_{\mathcal{C}} \models C_i$, $\mathcal{I}_{\mathcal{C}} \not\models D$ and $C \succ D$

Theorem

let $\mathcal{O}$ be sound and have the reduction property and let $\mathcal{C}$ be saturated wrt $\mathcal{O}$, then $\mathcal{C}$ is unsatisfiable iff $\mathcal{C}$ contains the empty clause

Theorem

let $\mathcal{O}$ be sound and have the reduction property and let $\mathcal{C}$ be saturated wrt $\mathcal{O}$, then $\mathcal{C}$ is unsatisfiable iff $\mathcal{C}$ contains the empty clause

Assumption

in the following we assume a language that contains $=$ as only predicate; for now we restrict to ground clauses

Theorem

let $\mathcal{O}$ be sound and have the reduction property and let $\mathcal{C}$ be saturated wrt $\mathcal{O}$, then $\mathcal{C}$ is unsatisfiable iff $\mathcal{C}$ contains the empty clause

Assumption

in the following we assume a language that contains $=$ as only predicate; for now we restrict to ground clauses

equality Herbrand interpretations are respresentable
by a convergent (wrt $\succ$) ground TRS

Theorem

let $\mathcal{O}$ be sound and have the reduction property and let $\mathcal{C}$ be saturated wrt $\mathcal{O}$, then $\mathcal{C}$ is unsatisfiable iff $\mathcal{C}$ contains the empty clause

Assumption

in the following we assume a language that contains $=$ as only predicate; for now we restrict to ground clauses

equality Herbrand interpretations are respresentable
by a convergent (wrt $\succ$) ground TRS

Definition

a clause $C \vee s = t$ is reductive if (i) $s \succ t$ and (ii) $s = t$ is strictly maximal wrt C

Theorem

let $\mathcal{O}$ be sound and have the reduction property and let $\mathcal{C}$ be saturated wrt $\mathcal{O}$, then $\mathcal{C}$ is unsatisfiable iff $\mathcal{C}$ contains the empty clause

Assumption

in the following we assume a language that contains $=$ as only predicate; for now we restrict to ground clauses

equality Herbrand interpretations are respresentable
by a convergent (wrt $\succ$) ground TRS

Definition

a clause $C \vee s = t$ is reductive if (i) $s \succ t$ and (ii) $s = t$ is strictly maximal wrt C

NB: a reductive clause may be viewed as a conditional rewrite rule, where negation is interpreted as non-derivability

let $\mathcal{C}_C = \{D \in \mathcal{C} \mid C \succ D\}$

let $\mathcal{C}_C = \{D \in \mathcal{C} \mid C \succ D\}$

Definition

we define a mapping $\mathcal{I}$ that assigns to $\forall\, \mathcal{C}_C$ a convergent TRS $\mathcal{I}_C$

let $\mathcal{C}_C = \{D \in \mathcal{C} \mid C \succ D\}$

Definition

we define a mapping $\mathcal{I}$ that assigns to $\forall\ \mathcal{C}_C$ a convergent TRS $\mathcal{I}_C$

$\mathcal{I}_C$ is the set of all ground rewrite rules $s \to t$ such that

1. $\exists\ D = C' \vee s = t \in \mathcal{C}$ with $C \succ D$

let $\mathcal{C}_C = \{D \in \mathcal{C} \mid C \succ D\}$

Definition

we define a mapping $\mathcal{I}$ that assigns to $\forall\ \mathcal{C}_C$ a convergent TRS $\mathcal{I}_C$

$\mathcal{I}_C$ is the set of all ground rewrite rules $s \to t$ such that

1. $\exists\ D = C' \vee s = t \in \mathcal{C}$ with $C \succ D$
2. D is reductive for $s = t$

let $\mathcal{C}_C = \{D \in \mathcal{C} \mid C \succ D\}$

Definition

we define a mapping $\mathcal{I}$ that assigns to $\forall\, \mathcal{C}_C$ a convergent TRS $\mathcal{I}_C$

$\mathcal{I}_C$ is the set of all ground rewrite rules $s \to t$ such that

1. $\exists\; D = C' \vee s = t \in \mathcal{C}$ with $C \succ D$
2. D is reductive for $s = t$
3. D is counter-example for $\mathcal{I}_D$

let $\mathcal{C}_C = \{D \in \mathcal{C} \mid C \succ D\}$

Definition

we define a mapping $\mathcal{I}$ that assigns to $\forall\ \mathcal{C}_C$ a convergent TRS $\mathcal{I}_C$

$\mathcal{I}_C$ is the set of all ground rewrite rules $s \to t$ such that

1. $\exists\ D = C' \vee s = t \in \mathcal{C}$ with $C \succ D$
2. D is reductive for $s = t$
3. D is counter-example for $\mathcal{I}_D$
4. s is in normal form wrt $\mathcal{I}_D$

let $\mathcal{C}_C = \{D \in \mathcal{C} \mid C \succ D\}$

Definition

we define a mapping $\mathcal{I}$ that assigns to $\forall\, \mathcal{C}_C$ a convergent TRS $\mathcal{I}_C$

$\mathcal{I}_C$ is the set of all ground rewrite rules $s \to t$ such that

1. $\exists\ D = C' \vee s = t \in \mathcal{C}$ with $C \succ D$
2. D is reductive for $s = t$
3. D is counter-example for $\mathcal{I}_D$
4. s is in normal form wrt $\mathcal{I}_D$
5. C' is counter-example for $\mathcal{I}_D \cup \{s = t\}$

let $\mathcal{C}_C = \{D \in \mathcal{C} \mid C \succ D\}$

Definition

we define a mapping $\mathcal{I}$ that assigns to $\forall\, \mathcal{C}_C$ a convergent TRS $\mathcal{I}_C$

$\mathcal{I}_C$ is the set of all ground rewrite rules $s \to t$ such that

1. $\exists\ D = C' \vee s = t \in \mathcal{C}$ with $C \succ D$
2. D is reductive for $s = t$
3. D is counter-example for $\mathcal{I}_D$
4. s is in normal form wrt $\mathcal{I}_D$
5. C' is counter-example for $\mathcal{I}_D \cup \{s = t\}$
6. we call D productive

let $\mathcal{C}_C = \{D \in \mathcal{C} \mid C \succ D\}$

Definition

we define a mapping $\mathcal{I}$ that assigns to $\forall\, \mathcal{C}_C$ a convergent TRS $\mathcal{I}_C$

$\mathcal{I}_C$ is the set of all ground rewrite rules $s \to t$ such that

1. $\exists\, D = C' \vee s = t \in \mathcal{C}$ with $C \succ D$
2. D is reductive for $s = t$
3. D is counter-example for $\mathcal{I}_D$
4. s is in normal form wrt $\mathcal{I}_D$
5. C' is counter-example for $\mathcal{I}_D \cup \{s = t\}$
6. we call D productive

Theorem

let $\mathcal{C}$ be a ground clause set; C a minimal counter-example to $\mathcal{I}_\mathcal{C}$;
$\exists\, D \in \mathrm{Res}_{\mathrm{SP}}(\mathcal{C})$ such that $C \succ D$ and D is also a counter-example

Redundancy and Saturation

Definitions

- a ground clause C is redundant wrt a ground clause set $\mathcal{C}$ if $\exists$ C_1, …, C_k in $\mathcal{C}$ such that

$$C_1, \dots, C_k \models C \qquad C \succ C_i$$

Redundancy and Saturation

Definitions

- a ground clause C is redundant wrt a ground clause set $\mathcal{C}$ if $\exists$ C_1, …, C_k in $\mathcal{C}$ such that

$$C_1, \dots, C_k \models C \qquad C \succ C_i$$

- a ground inference

$$\frac{C_1 \quad \dots \quad C_n \quad C}{D}$$

  is redundant (wrt $\mathcal{C}$) if

  1. C main premise
  2. $D \succcurlyeq C$, or
  3. $\exists$ $D_1, \dots, D_k$ with $D_i \in \mathcal{C}_C$ such that $D_1, \dots, D_k, C_1, \dots, C_n \models D$

Redundancy and Saturation

Definitions

- a ground clause C is redundant wrt a ground clause set $\mathcal{C}$ if $\exists\ C_1$, $\dots$, C_k in $\mathcal{C}$ such that

$$C_1, \dots, C_k \models C \qquad C \succ C_i$$

- a ground inference

$$\frac{C_1 \quad \dots \quad C_n \quad C}{D}$$

  is redundant (wrt $\mathcal{C}$) if

  1. C main premise
  2. $D \succcurlyeq C$, or
  3. $\exists\ D_1, \dots, D_k$ with $D_i \in \mathcal{C}_C$ such that $D_1, \dots, D_k, C_1, \dots, C_n \models D$

- $\mathcal{C}$ is saturated upto redundancy if all inferences from non-redundant premises are redundant

Soundness and Completeness of Superposition

Theorem

let $\mathcal{O}$ be sound and have the reduction property and let $\mathcal{C}$ be saturated upto redundancy wrt $\mathcal{O}$, then $\mathcal{C}$ is unsatisfiable iff $\mathcal{C}$ contains the empty clause

Soundness and Completeness of Superposition

Theorem

let $\mathcal{O}$ be sound and have the reduction property and let $\mathcal{C}$ be saturated upto redundancy wrt $\mathcal{O}$, then $\mathcal{C}$ is unsatisfiable iff $\mathcal{C}$ contains the empty clause

Lemma

non-redundant superposition inferences are liftable

Proof.

on the whiteboard

Soundness and Completeness of Superposition

Theorem

let $\mathcal{O}$ be sound and have the reduction property and let $\mathcal{C}$ be saturated upto redundancy wrt $\mathcal{O}$, then $\mathcal{C}$ is unsatisfiable iff $\mathcal{C}$ contains the empty clause

Lemma

non-redundant superposition inferences are liftable

Proof.

on the whiteboard ■

Soundness and Completeness of Superposition

Theorem

let $\mathcal{O}$ be sound and have the reduction property and let $\mathcal{C}$ be saturated upto redundancy wrt $\mathcal{O}$, then $\mathcal{C}$ is unsatisfiable iff $\mathcal{C}$ contains the empty clause

Lemma

non-redundant superposition inferences are liftable

Proof.

on the whiteboard ∎

Theorem

superposition is sound and complete; let F be a sentence and $\mathcal{C}$ its clause form; then F is unsatisfiable iff $\square \in \mathsf{Res_{SP}}^*(\mathcal{C})$