Computational Logic

Learning-Assisted Automated Reasoning

Lecture 2

Cezary Kaliszyk

October 19, 2018

Summary

Last time

Theorem proving systems

Machine learning problems

Lemma relevance

Features and deep approaches

Today

ATP Calculi

Resolution

Orderings

Reasoning with equality

Learning the foundations

Slides include some material from the LICS and ATP courses by Aart Middeldorp and Georg Moser taught at the University of Innsbruck

Calculi and approaches

- Propositional logic: SAT
- Description logic and variants: tableaux
- First-order logic: (ordered) resolution, tableaux
- FOL with equality: superposition
- Higher-order logic: Satallax, Leo-III, ...
- Curry-Howard based systems: fragments or proof term generation

Definitions

- clause is set of literals $\{l_1, \ldots, l_n\}$ representing formula $l_1 \vee \cdots \vee l_n$
- clausal form is set of clauses $\{C_1, \ldots, C_m\}$ representing formula $C_1 \wedge \cdots \wedge C_m$

Corollary

$\forall$ *formula* ϕ $\exists$ *clausal form* ψ *such that* $\phi \equiv \psi$

Examples

CNF	clausal form
$\neg p \wedge (\neg q \vee \neg p) \wedge (\neg p \vee \neg r)$	$\{\{\neg p\}, \{\neg q, \neg p\}, \{\neg p, \neg r\}\}$
$(\neg p \vee q) \wedge (q \vee \neg r) \wedge (p \vee q \vee \neg r)$	$\{\{\neg p, q\}, \{q, \neg r\}, \{p, q, \neg r\}\}$
$(\neg p \vee \neg p) \wedge (q \vee r) \wedge (r \vee q)$	$\{\{\neg p\}, \{q, r\}\}$

Definition

literals l_1 and l_2 are complementary if $\underbrace{l_1 = \neg l_2 \text{ or } \neg l_1 = l_2}_{l_1 = l_2^c}$

Notation

if l is literal then $l^c = \begin{cases} \neg p & \text{if } l = p \\ p & \text{if } l = \neg p \end{cases}$

Definition

- clauses C_1 and C_2 clash if $\exists$ literal l with $l \in C_1$ and $l^c \in C_2$
- resolvent of clashing clauses C_1 and C_2 is clause

$$(C_1 \setminus \{l\}) \cup (C_2 \setminus \{l^c\})$$

- C_1 and C_2 are parent clauses of resolvent

Resolution

input: clausal form S

output: yes if S is satisfiable
no if S is unsatisfiable

1. repeatedly add resolvent of pair of clashing clauses in S
2. return no as soon as empty clause $\square$ is derived
3. return yes if all clashing clauses have been resolved

Definition

refutation of S is resolution derivation of $\square$ from S

Theorem

resolution is sound and complete:

clausal form S is unsatisfiable if and only if S admits refutation

Example (1)

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

1. $\{\neg p, \neg q, r\}$
2. $\{p, r\}$
3. $\{q, r\}$
4. $\{\neg r\}$
5. $\{\neg q, r\}$ resolve 1, 2
6. $\{r\}$ resolve 3, 5
7. $\square$ resolve 4, 6

unsatisfiable

First-order case

Definition

- literal is atom $P(t_1, \ldots, t_n)$ or negation of atom $\neg P(t_1, \ldots, t_n)$
- clause is set of literals $\{l_1, \ldots, l_n\}$, representing $l_1 \vee \cdots \vee l_n$
- clausal form is set of clauses $\{C_1, \ldots, C_m\}$, representing $\forall (C_1 \wedge \cdots \wedge C_m)$
- clauses C_1 and C_2 without common variables clash if $\exists$ literals $l_1 \in C_1$ and $l_2 \in C_2$ such that l_1 and l_2^c are unifiable
- binary resolvent of clashing clauses C_1 and C_2 is clause

$$((C_1 \setminus \{l_1\}) \cup (C_2 \setminus \{l_2\}))\theta$$

where θ is mgu of l_1 and l_2^c

Example (1)

1.	$\{\neg P(x), Q(x), R(x, f(x))\}$		
2.	$\{\neg P(x), Q(x), S(f(x))\}$		
3.	$\{T(a)\}$		
4.	$\{P(a)\}$		
5.	$\{\neg R(a, y), T(y)\}$		
6.	$\{\neg T(x), \neg Q(x)\}$		
7.	$\{\neg T(x), \neg S(x)\}$		
8.	$\{\neg Q(a)\}$	resolve 3, 6	$\{a/x\}$
9.	$\{Q(a), S(f(a))\}$	resolve 2, 4	$\{a/x\}$
10.	$\{Q(a), R(a, f(a))\}$	resolve 1, 4	$\{a/x\}$
11.	$\{S(f(a))\}$	resolve 8, 9	
12.	$\{R(a, f(a))\}$	resolve 8, 10	
13.	$\{T(f(a))\}$	resolve 5, 12	$\{f(a)/y\}$
14.	$\{\neg S(f(a))\}$	resolve 7, 13	$\{f(a)/x\}$
15.	$\square$	resolve 11,14	

Theorem

binary resolution is sound:

clausal form S is unsatisfiable if S admits refutation

Problem

binary resolution is incomplete

Example

1. $\{P(x), P(y)\}$
2. $\{\neg P(x'), \neg P(y')\}$
3. $\{P(y), \neg P(y')\}$ resolve 1, 2 $\{x'/x\}$

unsatisfiable but no refutation

Solution

incorporate factoring: $C\sigma$ is a factor of C if two or more literals in C have mgu σ

Example

1. $\{P(x), P(y)\}$ factor $\{P(x)\}$
2. $\{\neg P(x'), \neg P(y')\}$ factor $\{\neg P(x')\}$
3. $\square$

refutation

First-Order Resolution

Definition

resolution

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

factoring

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

σ is a mgu of the atomic formulas A and B

Theorem

resolution is sound and complete:

clausal form S is unsatisfiable if and only if S admits refutation

We can now build a prover based on resolution. But will it be efficient?

Orders

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

Example: KBOs

Definitions

$w(x) = w_0 \geqslant 1$

- let $\succ$ denote a precedence on some signature
- let w denote an admissible weight function into natural numbers

Definition

extension of w to terms

$s \succ_{\text{kbo}} t$ if $\forall\ x$: $|s|_x \geqslant |t|_x$ and

1. $\text{weight}(s) > \text{weight}(t)$, or
2. $\text{weight}(s) = \text{weight}(t)$, and one of the following alternatives holds:
   - 2.1 t is a variable, $s = f^n(t)$, $n > 0$,
   - 2.2 $s = f(s_1, \ldots, s_n)$, $t = f(t_1, \ldots, t_n)$,
     $\exists\ i$ such that $s_i \succ_{\text{kbo}} t_i\ \forall\ 1 \leqslant j < i$: $s_j = t_j$
   - 2.3 $s = f(s_1, \ldots, s_n)$, $t = g(t_1, \ldots, t_m)$, $f \succ g$.

Ordered Resolution Calculus

σ is ground if $E\sigma$ is ground

Definition

- a literal L is maximal if $\exists$ ground σ such that for no other literal M: $M\sigma \succ_{\mathrm{L}} L\sigma$
- L is strictly maximal if $\exists$ ground σ such that for no other literal M: $M\sigma \succcurlyeq_{\mathrm{L}} L\sigma$; here $\succcurlyeq_{\mathrm{L}}$ 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 $D\sigma$

Subsumption

Lemma

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

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

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

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

Paramodulation Calculus

Definition

- let $\Box$ be a fresh constant; let $\mathcal{L}$ be our basic language
- terms of $\mathcal{L} \cup \{\Box\}$ such that $\Box$ occurs exactly once, are called contexts
- empty context is denoted as $\Box$
- for context $C[\Box]$ and a term t
  we write $C[t]$ for the replacement of $\Box$ by t

Example

- let $\mathcal{L} = \{\mathsf{c}, \mathsf{f}, \mathsf{P}\}$
- $\mathsf{P}(\mathsf{f}(\Box)) =: C[\Box]$ is a context
- $C[\mathsf{f}(\mathsf{c})] = \mathsf{P}(\mathsf{f}(\mathsf{f}(\mathsf{c})))$

Definition

$$\frac{C \vee A \quad D \vee \neg B}{(C \vee D)\sigma_1} \qquad \frac{C \vee A \vee B}{(C \vee A)\sigma_1}$$

$$\frac{C \vee s \neq s'}{C\sigma_2} \qquad \frac{C \vee s = t \quad D \vee L[s']}{(C \vee D \vee L[t])\sigma_2}$$

- σ_1 is a mgu of A and B (A, B atomic)
- σ_2 is a mgu of s and s'

Example

consider $\mathcal{C} = \{\mathsf{c} \neq \mathsf{d}, \mathsf{b} = \mathsf{d}, \mathsf{a} \neq \mathsf{d} \vee \mathsf{a} = \mathsf{c}, \mathsf{a} = \mathsf{b} \vee \mathsf{a} = \mathsf{d}\}$

$$\dfrac{\dfrac{\dfrac{\dfrac{\mathsf{b} = \mathsf{d} \quad \mathsf{a} = \mathsf{b} \vee \mathsf{a} = \mathsf{d}}{\mathsf{a} = \mathsf{d} \vee \mathsf{a} = \mathsf{d}}}{\mathsf{a} = \mathsf{d}} \qquad \mathsf{c} \neq \mathsf{d}}{\mathsf{a} \neq \mathsf{c}} \qquad \dfrac{\dfrac{\mathsf{a} = \mathsf{d} \quad \mathsf{a} \neq \mathsf{d} \vee \mathsf{a} = \mathsf{c}}{\mathsf{d} \neq \mathsf{d} \vee \mathsf{a} = \mathsf{c}}}{\mathsf{a} = \mathsf{c}}}{\square}$$

Ordered Paramodulation Calculus

Definition

$$\frac{C \vee A \quad D \vee \neg B}{(C \vee D)\sigma_1} \qquad \frac{C \vee A \vee B}{(C \vee A)\sigma_1}$$

$$\frac{C \vee s \neq s'}{C\sigma_2} \qquad \frac{C \vee s = t \quad D \vee L[s']}{(C \vee D \vee L[t])\sigma_2}$$

- same conditions on σ_1, σ_2 as before
- $A\sigma_1$ is strictly maximal with respect to $C\sigma_1$; $\neg B\sigma_1$ is maximal with respect to $D\sigma_1$
- the equation $(s = t)\sigma_2$ and the literal $L[s']\sigma_2$ are maximal with respect to $D\sigma_2$

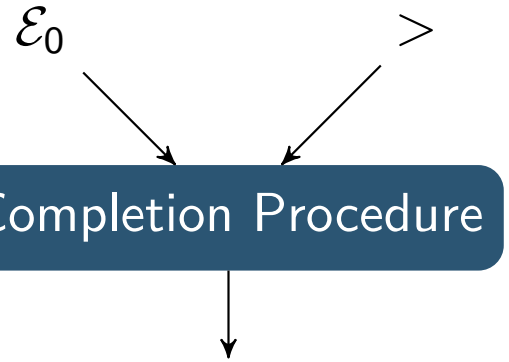
Theorem

ordered paramodulation is sound and complete

But we still deal with equalities in an unrestricted way!

Completion

Input: Initial set of equations and a reduction order



$$(\mathcal{E}_0, \varnothing) \vdash (\mathcal{E}_1, \mathcal{R}_1) \vdash (\mathcal{E}_2, \mathcal{R}_2) \vdash (\mathcal{E}_3, \mathcal{R}_3) \vdash \cdots$$

deduce $\dfrac{(\mathcal{E}, \mathcal{R})}{(\mathcal{E} \cup \{s \approx t\}, \mathcal{R})}$ if $s \;{}_{\mathcal{R}}\!\leftarrow u \rightarrow_{\mathcal{R}} t$

orient $\dfrac{(\mathcal{E} \cup \{s \mathrel{\dot\approx} t\}, \mathcal{R})}{(\mathcal{E}, \mathcal{R} \cup \{s \to t\})}$ if $s > t$

simplify $\dfrac{(\mathcal{E} \cup \{s \mathrel{\dot\approx} t\}, \mathcal{R})}{(\mathcal{E} \cup \{u \mathrel{\dot\approx} t\}, \mathcal{R})}$ if $s \rightarrow_{\mathcal{R}} u$

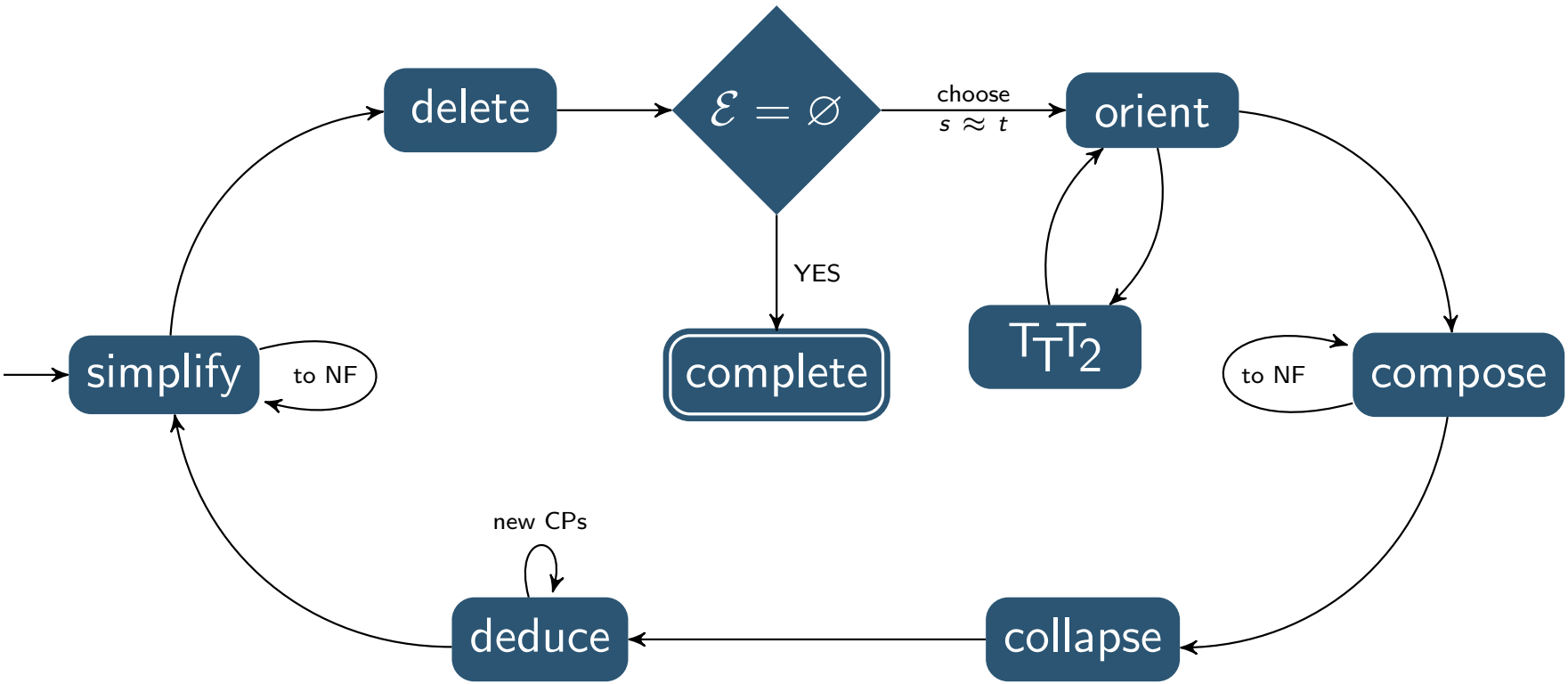
delete $\dfrac{(\mathcal{E} \cup \{s \approx s\}, R)}{(\mathcal{E}, \mathcal{R})}$

compose $\dfrac{(\mathcal{E}, \mathcal{R} \cup \{s \to t\})}{(\mathcal{E}, \mathcal{R} \cup \{s \to u\})}$ if $t \rightarrow_{\mathcal{R}} u$

collapse $\dfrac{(\mathcal{E}, \mathcal{R} \cup \{s \to t\})}{(\mathcal{E} \cup \{u \approx t\}, \mathcal{R})}$ if $s \overset{!}{\rightarrow}_{\mathcal{R}} u$

Knuth-Bendix Completion

- Knuth-Bendix completion in term rewriting research
- Reduction order derived on the fly



Superposition Calculus

Definition

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

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

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

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

- 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))

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

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

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

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

constraints:

1. for the superposition rules: σ is a mgu of s and s', s' not a variable, $t\sigma \not\succcurlyeq s\sigma$, $v\sigma \not\succcurlyeq u[s']\sigma$, $(s = t)\sigma$ 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. $(s = t)\sigma \not\succcurlyeq (u[s'] = v)\sigma$

Redundancy and Saturation

Definitions

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

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

- a ground inference with main premise C

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

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

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

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