

Lecture 2: Reasoning with FOL

2-AIN-108 Computational Logic

Martin Baláž, Martin Homola

Department of Applied Informatics
Faculty of Mathematics, Physics and Informatics
Comenius University in Bratislava



1 Oct 2013

Intuitions:

- 1 Formula $P \wedge (P \rightarrow Q) \rightarrow Q$ is a tautology.
- 2 Hence for any theory T : if $T \models P$ and $T \models P \rightarrow Q$ we can conclude $T \models Q$.
- 3 We express this with with the **derivation rule** Modus Ponens:

$$\frac{P, P \rightarrow Q}{Q}$$

Note: A tautology is a formula that is satisfied by any first order structure. A contradiction is a formula that is unsatisfiable.

- 1 **Calculus** is a system which allows us to derive formulae by derivation rules.
- 2 Derivation of a formula Φ from T is called a **proof** of Φ from T .
- 3 We denote by $T \vdash \phi$ if formula Φ is derived from T by the calculus.

Definition (Soundness)

A calculus is **sound** iff for all theories T and for all formulae Φ , $T \vdash \Phi$ implies $T \models \Phi$.

Definition (Completeness)

A calculus is **complete** iff for all theories T and for all formulae Φ , $T \models \Phi$ implies $T \vdash \Phi$.

Definition (Substitution)

The formula resulting from Φ by **substitution** of a variable x by some term t (denoted $\Phi\{x/t\}$) is a formula Ψ identical to Φ except for every occurrence of x is replaced by t .

A term t is **substitutable** for a variable x in a formula Φ iff no occurrence of a variable in t becomes bounded after the substitution.

Definition (Substitution)

The formula resulting from Φ by **substitution** of a variable x by some term t (denoted $\Phi\{x/t\}$) is a formula Ψ identical to Φ except for every occurrence of x is replaced by t .

A term t is **substitutable** for a variable x in a formula Φ iff no occurrence of a variable in t becomes bounded after the substitution.

Intuition: x is not substitutable for y in the following example:

$$\begin{aligned}\Phi &= (\exists x)(y < x) \\ \Phi\{y/x\} &= (\exists x)(x < x)\end{aligned}$$

Axioms

- 1 $(P \rightarrow (Q \rightarrow P))$
- 2 $((P \rightarrow (Q \rightarrow R)) \rightarrow ((P \rightarrow Q) \rightarrow (P \rightarrow R)))$
- 3 $((\neg P \rightarrow \neg Q) \rightarrow (Q \rightarrow P))$
- 4 $((\forall x)P \rightarrow P\{x/t\})$
where term t is substitutable for x in P
- 5 $((\forall x)(P \rightarrow Q) \rightarrow (P \rightarrow (\forall x)Q))$
where x does not occur free in P

Inference Rules

- Modus Ponens (MP):

$$\frac{P, (P \rightarrow Q)}{Q}$$

- Generalization (G):

$$\frac{P}{(\forall x)P}$$

Hilbert Calculus (cont.)

A proof of Φ from T in Hilbert Calculus is a sequence $\langle \Phi_1, \Phi_2, \dots, \Phi_n \rangle$ s.t. $\Phi_n = \Phi$ and for all $1 \leq i \leq n$ one of the following holds:

- 1 Φ_i instantiates an axiom;
- 2 $\Phi_i \in T$;
- 3 Φ_i is derived from the formulae $\Phi_1, \dots, \Phi_{i-1}$ by one of the derivation rules.

We write $T \vdash \Phi$ if there exists a proof from of Φ from T .

Example

Prove:

$$(P(t) \rightarrow (\exists x)P(x)) \quad \text{i.e.} \quad (P(t) \rightarrow \neg(\forall x)\neg P(x))$$

where t is substitutable for x in P .

Example

Prove:

$$(P(t) \rightarrow (\exists x)P(x)) \quad \text{i.e.} \quad (P(t) \rightarrow \neg(\forall x)\neg P(x))$$

where t is substitutable for x in P .

Proof:

- 1 $((\forall x)\neg P(x) \rightarrow \neg P(t))$ (Axiom 4)
- 2 $((\forall x)\neg P(x) \rightarrow \neg P(t)) \rightarrow (P(t) \rightarrow \neg(\forall x)\neg P(x))$ (Axiom 3)
- 3 $(P(t) \rightarrow \neg(\forall x)\neg P(x))$ (MP)

Theorem (Soundness & completeness)

Hilbert calculus for FOL is sound and complete.

Resolution

Definition

A **literal** is either an atom or an atom preceded by negation (\neg).

Definition

A **clause** is a disjunction of literals.

Definition

A **literal** is either an atom or an atom preceded by negation (\neg).

Definition

A **clause** is a disjunction of literals.

Example: Which of the following formulae are clauses?

$$P(x) \vee \neg Q(x) \quad (1)$$

$$P(x) \vee Q(x) \wedge S(x, y) \quad (2)$$

$$(\exists x)P(x) \quad (3)$$

$$(\forall x)(\neg P(x) \vee Q(x)) \quad (4)$$

Definition

A **literal** is either an atom or an atom preceded by negation (\neg).

Definition

A **clause** is a disjunction of literals.

Example: Which of the following formulae are clauses?

$$P(x) \vee \neg Q(x) \quad (1)$$

$$P(x) \vee Q(x) \wedge S(x, y) \quad (2)$$

$$(\exists x)P(x) \quad (3)$$

$$(\forall x)(\neg P(x) \vee Q(x)) \quad (4)$$

Note: we will understand clauses as closed, universally quantified formulae, but we will omit the quantifiers.

Definition (Complementary literals)

Given any atom A , we say that the two literals A and $\neg A$ are **complementary**.

Intuition: (Simplified) resolution rule:

$$\frac{P \vee Q, \neg P \vee R}{Q \vee R} \quad \frac{Q \vee P, R \vee \neg P}{Q \vee R}$$

Note: we say that the two clauses $P \vee Q$ and $\neg P \vee R$ ($R \vee \neg P$) containing complementary literals P and $\neg P$ resolve into the single clause $Q \vee R$.

Negation Normal Form

Definition (Negation normal form)

A formula ϕ is in the **negation normal form** (NNF) iff $\{\neg, \wedge, \vee\}$ are the only allowed connectives and negation only occurs in front of atoms in ϕ .

Definition (Negation normal form)

A formula ϕ is in the **negation normal form** (NNF) iff $\{\neg, \wedge, \vee\}$ are the only allowed connectives and negation only occurs in front of atoms in ϕ .

Transform any formula into NNF:

- Double negative law:

$$\neg\neg P / P$$

- De Morgan's law:

$$\neg(P \wedge Q) / (\neg P \vee \neg Q)$$

$$\neg(P \vee Q) / (\neg P \wedge \neg Q)$$

- Quantifiers:

$$\neg(\forall x)P / (\exists x)\neg P$$

$$\neg(\exists x)P / (\forall x)\neg P$$

Definition (Prenex normal form)

A formula is in **prenex normal form** (PNF) iff it is of the form $(Q_1x_1) \dots (Q_nx_n)F$, $n \geq 0$, where Q_i is a quantifier, x_i is a variable and F is quantifier-free formula.

Definition (Prenex normal form)

A formula is in **prenex normal form** (PNF) iff it is of the form $(Q_1x_1) \dots (Q_nx_n)F$, $n \geq 0$, where Q_i is a quantifier, x_i is a variable and F is quantifier-free formula.

Transform a **formula in NNF** into PNF – push quantifiers outwards:

- Conjunction:

$$((\forall x)P \wedge Q)/(\forall x)(P \wedge Q) \quad (Q \wedge (\forall x)P)/(\forall x)(Q \wedge P)$$

$$((\exists x)P \wedge Q)/(\exists x)(P \wedge Q) \quad (Q \wedge (\exists x)P)/(\exists x)(Q \wedge P)$$

if x does not appear as free variable in Q

- Disjunction:

$$((\forall x)P \vee Q)/(\forall x)(P \vee Q) \quad (Q \vee (\forall x)P)/(\forall x)(Q \vee P)$$

$$((\exists x)P \vee Q)/(\exists x)(P \vee Q) \quad (Q \vee (\exists x)P)/(\exists x)(Q \vee P)$$

if x does not appear as free variable in Q

Definition (Skolem normal form)

A formula is in **Skolem normal form** (SNF) iff it is in PNF with only universal quantifiers.

Definition (Skolem normal form)

A formula is in **Skolem normal form** (SNF) iff it is in PNF with only universal quantifiers.

Skolemize a formula in PNF:

- 1 Given $\Phi = (\forall x_1) \dots (\forall x_n) (\exists y) \Psi$, replace $(\exists y) \Psi$ with Ψ' in which every occurrence of y is replaced by $f(x_1, \dots, x_n)$ where f is a new function symbol.
- 2 Repeat until there are no existential quantifiers.

Definition (Skolem normal form)

A formula is in **Skolem normal form** (SNF) iff it is in PNF with only universal quantifiers.

Skolemize a formula in PNF:

- 1 Given $\Phi = (\forall x_1) \dots (\forall x_n) (\exists y) \Psi$, replace $(\exists y) \Psi$ with Ψ' in which every occurrence of y is replaced by $f(x_1, \dots, x_n)$ where f is a new function symbol.
- 2 Repeat until there are no existential quantifiers.

Note: Φ and the resulting formula Φ' are equisatisfiable (i.e., one is satisfiable iff the other one is). They are not necessarily equivalent.

Note: the new function f is called Skolem function. If f is nullary, it is called Skolem constant.

Conjunctive Normal Form

Definition (Conjunctive normal form)

A formula is in **conjunctive normal form** (CNF) iff it is a conjunction of clauses.

Definition (Conjunctive normal form)

A formula is in **conjunctive normal form** (CNF) iff it is a conjunction of clauses.

Transform Φ into CNF:

- 1 Negation Normal Form
- 2 Prenex Normal Form
- 3 Skolem Normal Form
- 4 Apply distributive law:

$$((P \wedge Q) \vee R) / ((P \vee R) \wedge (Q \vee R))$$

$$(P \vee (Q \wedge R)) / ((P \vee Q) \wedge (P \vee R))$$

Definition (Unification)

Given two literals P, Q and a substitution θ , we say that $\text{Unify}(P, Q, \theta)$ is true if $P\theta = Q\theta$.

Resolution rule:

$$\frac{P_1 \vee \dots \vee P_i \vee \dots \vee P_m, Q_1 \vee \dots \vee Q_j \vee \dots \vee Q_n, \text{Unify}(P_i, \neg Q_j, \theta)}{P_1 \vee \dots \vee P_{i-1} \vee P_{i+1} \vee \dots \vee P_m \vee Q_1 \vee \dots \vee Q_{j-1} \vee Q_{j+1} \vee \dots \vee Q_n \theta}$$

where for all k, l : P_k, Q_l are literals.

Theorem

Given a first order theory T and any formula ϕ we have: $T \models \phi$ iff $T \cup \{\neg\phi\}$ is unsatisfiable.

Algorithm: Resolution

Input: FOL theory T , formula ϕ

Output: True if $T \models \phi$

- 1 Transform $T \cup \{\neg\phi\}$ into CNF, yielding a set of clauses.
- 2 Exhaustively apply the resolution rule to all possible clauses that contain complementary literals
 - all repeated literals are removed
 - all clauses with complementary literals are discarded
- 3 if *empty clause* is derived answer “True” $T \wedge \neg\phi$ is not satisfiable; answer “False” if it is not possible to resolve any more clauses.

Theorem (Soundness & completeness)

The resolution algorithm is sound and complete

Theorem (Soundness & completeness)

The resolution algorithm is sound and complete, i.e., given input T and Φ , if the algorithm answers “True” then $T \models \Phi$ (soundness)

Theorem (Soundness & completeness)

The resolution algorithm is sound and complete, i.e., given input T and Φ , if the algorithm answers “True” then $T \models \Phi$ (soundness), and, if $T \models \Phi$ the algorithm answers “True” (completeness).

Theorem (Soundness & completeness)

The resolution algorithm is sound and complete, i.e., given input T and Φ , if the algorithm answers “True” then $T \models \Phi$ (soundness), and, if $T \models \Phi$ the algorithm answers “True” (completeness).

Theorem (Termination)

If $T \models \Phi$ then the resolution algorithm eventually terminates, given T and Φ on input.

Theorem (Soundness & completeness)

The resolution algorithm is sound and complete, i.e., given input T and Φ , if the algorithm answers “True” then $T \models \Phi$ (soundness), and, if $T \models \Phi$ the algorithm answers “True” (completeness).

Theorem (Termination)

If $T \models \Phi$ then the resolution algorithm eventually terminates, given T and Φ on input.

Note: the resolution algorithm may not terminate, if $T \not\models \Phi$ – the algorithm is semidecidable.