Computational Logic

First-Order Logic

Martin Baláž

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

2011

Alphabet

An *alphabet* contains

- Variables
  $x, y, z, \ldots$
- Constants
  $c, d, e, \ldots$
- Function symbols
  $f, g, h, \ldots$
- Predicate symbols
  $p, q, r, \ldots$
- Logical connectives
  $\neg, \vee, \wedge, \Rightarrow, \Leftrightarrow, \ldots$
- Quantifiers
  $\forall \quad \exists$
- Punctuation symbols
  $(\quad) \quad ,$

Term and Atom

A *term* is

- a variable
- a constant
- an expression $f(t_1, \ldots, t_n)$ if f is a function symbol with arity n and $t_1, \ldots, t_n$ are terms

A *atom* is an expression $p(t_1, \ldots, t_n)$ where p is a predicate symbol with arity n and $t_1, \ldots, t_n$ are terms.

Formula

A *formula* is

- an atom
- $\neg\Phi$ if Φ is a formula
- $(\Phi \wedge \Psi)$ if Φ and Ψ are formulas
- $(\Phi \vee \Psi)$ if Φ and Ψ are formulas
- $(\Pi \rightarrow \Psi)$ if Φ and Ψ are formulas
- $(\Pi \leftrightarrow \Psi)$ if Φ and Ψ are formulas
- ...
- $(\forall x)\Phi$ if x is a variable and Φ is a formula
- $(\exists x)\Phi$ if x is a variable and Φ is a formula

A *language* is a set $\mathcal{L}$ of all formulas.

Structure

A *domain* is a set of individuals D.
A *signature* is a tripple $\sigma = (F, P, arity)$ where

- F is a set of function symbols
- P is a set of predicate symbols
- $arity : F \cup P \mapsto N$ is an arity function

An *interpretation* is a function I such that

- $I(f)$ is a function $f^I : D^{arity(f)} \mapsto D$
- $I(p)$ is a relation $p^I \subseteq D^{arity(p)}$

A *structure* is a tripple $\mathcal{D} = (D, \sigma, I)$ where

- D is a domain
- σ is a signature
- I is an interpretation function

Example

$$(\forall x)p(c,x,x)$$
$$(\forall x)(\forall y)(\forall z)(p(x,g(y),z) \Leftrightarrow p(f(x),y,z))$$

- Domain $D = N$
- Signature
  $\sigma = (\{c,f,g\},\{p\},\{c \mapsto 0, f \mapsto 1, g \mapsto 1, p \mapsto 3\})$
- Interpretation
  $I(c) = 0$
  $I(f) = x \mapsto x+1$
  $I(g) = x \mapsto x+1$
  $I(p) = \{(x,y,z) \mid x+y=z\}$

Variable Assignment

A *variable assignment* is a mapping $e\colon X \mapsto D$ where X is a set of variables and D is a domain.
If $x \in X$ is a variable and $d \in D$ is an individual, then by $e(x \mapsto d)$ we will denote a variable assignment satisfying

$$e(x \mapsto d)(y) = \begin{cases} d & \text{if } x = y \\ e(y) & \text{if } x \neq y \end{cases}$$

Let $\mathcal{D}$ be a struture and e be a variable assignment.
The *value of* a term t (denoted by $t[e]$) is

- $e(t)$ if t is a variable
- c^I if t is a constant
- $f^I(t_1[e], \dots, t_n[e])$ if $t = f(t_1, \dots, t_n)$ is a compound term

A formula Φ is *true* w.r.t. $\mathcal{D}$ and e (denoted by $\mathcal{D} \models \Phi[e]$) iff

- $\mathcal{D} \models p(t_1, \dots, t_n)[e]$ iff $(t_1[e], \dots, t_n[e]) \in p^I$
- $\mathcal{D} \models \neg\Phi[e]$ iff $\mathcal{D} \not\models \Phi[e]$
- $\mathcal{D} \models (\Phi \wedge \Psi)[e]$ iff $\mathcal{D} \models \Phi[e]$ and $\mathcal{D} \models \Psi[e]$
- $\mathcal{D} \models (\Phi \vee \Psi)[e]$ iff $\mathcal{D} \models \Phi[e]$ or $\mathcal{D} \models \Psi[e]$
- $\mathcal{D} \models (\Phi \rightarrow \Psi)[e]$ iff $\mathcal{D} \not\models \Phi[e]$ or $\mathcal{D} \models \Psi[e]$
- $\mathcal{D} \models (\Phi \leftrightarrow \Psi)[e]$ iff $\mathcal{D} \models \Phi[e]$ iff $\mathcal{D} \models \Psi[e]$
- $\mathcal{D} \models (\forall x)\Phi[e]$ iff $\mathcal{D} \models \Phi[e(x \mapsto d)]$ for all $d \in D$
- $\mathcal{D} \models (\exists x)\Phi[e]$ iff $\mathcal{D} \models \Phi[e(x \mapsto d)]$ for some $d \in D$

Entailment

A formula Φ is *true* w.r.t. a structure $\mathcal{D}$ (denoted by $\mathcal{D} \models \Phi$) iff $\mathcal{D} \models \Phi[e]$ for all variable assignments e.
A set of formulas T *entails* a formula Φ (denoted by $T \models \Phi$) iff for all structures $\mathcal{D}$ holds $\mathcal{D} \models \Phi$ whenever $\mathcal{D} \models \Psi$ for all Ψ in T.

Substitution

A *free occurence of a variable* is not bounded with a quantifier.

A *substitution* $\Phi(x/t)$ means replace every occurence of the variable x in the formula Φ with the term t.

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

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

Normal Forms

A formula is in *negation normal form* iff if $\{\neg, \wedge, \vee\}$ are are the only allowed connectives and literals are the only negated subformulas.

A formula is in *prenex normal form* iff it is of the form $(Q_1 x_1) \dots (Q_n x_n) F$, $n \geq 0$, where Q_i is a quantifier, x_i is a variable and F is quantifier-free formula.

A formula is in *Skolem normal form* iff it is in prenex normal form with only universal quantifiers.

A formula is in *conjunctive normal form* iff it is conjunction of disjunctive clauses, where a *disjunctive clause* is a disjunction of literals.

A formula is in *disjunctive normal form* iff it is disjunction of conjunctive clauses, where a *conjunctive clause* is a conjunction of literals.

Negation Normal Form

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

Prenex Normal Form

- Negation:
  $\neg(\exists x)P / (\forall x)\neg P$
  $\neg(\forall x)P / (\exists x)\neg P$
- 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
- Implication:
  $((\forall x)P \rightarrow Q)/(\exists x)(P \rightarrow Q) \quad (Q \rightarrow (\forall x)P)/(\forall x)(Q \rightarrow P)$
  $((\exists x)P \rightarrow Q)/(\forall x)(P \rightarrow Q) \quad (Q \rightarrow (\exists x)P)/(\exists x)(Q \rightarrow P)$
  if x does not appear as free variable in Q

Skolem Normal Form

Formulas P and Q are *equisatisfiable* if P is satisfiable if and only if Q is satisfiable.

Given a formula F:

1. If F is already in Skolem normal form, we are done.
2. If not, then F is of the form

   $$(\forall x_1)\dots(\forall x_m)(\exists y)F'(x_1,\dots,x_m,y,z_1,\dots,z_n)$$

   where each z_i is a free variable and F' is in prenex normal form. Replace y with $f(x_1,\dots,x_m,z_1,\dots,z_n)$ where f is a new function symbol.
3. go to 1

Conjunctive Normal Form

1. Negation Normal Form
2. Prenex Normal Form
3. Skolem Normal Form
4. Distributive law ($\vee$ over $\wedge$):
   $((P \wedge Q) \vee R)/((P \vee R) \wedge (Q \vee R))$
   $(P \vee (Q \wedge R))/((P \vee Q) \wedge (P \vee R))$

Disjunctive Normal Form

1. Negation Normal Form
2. Prenex Normal Form
3. Skolem Normal Form
4. Distributive law ($\wedge$ over $\vee$):
   $((P \vee Q) \wedge R)/((P \wedge R) \vee (Q \wedge R))$
   $(P \wedge (Q \vee R))/((P \wedge Q) \vee (P \wedge R))$

Hilbert System

Axioms

- $(P \to (Q \to P))$
- $((P \to (Q \to R)) \to ((P \to Q) \to (P \to R)))$
- $((\neg P \to \neg Q) \to (Q \to P))$
- $((\forall x)P \to P(x/t))$
  where term t is substitutable for x in P
- $((\forall x)(P \to Q) \to (P \to (\forall x)Q))$
  where x does not occur free in P

Inference Rules

- Modus Ponens

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

- Generalization

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

Example

Prove:

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

where t is substitutable for x in P.

Proof:

1. $((\forall x)\neg P \rightarrow \neg P(x/t))$
2. $(((\forall x)\neg P \rightarrow \neg P(x/t)) \rightarrow (P(x/t) \rightarrow \neg(\forall x)\neg P))$
   Note: $((A \rightarrow \neg B) \rightarrow (B \rightarrow \neg A))$ is a tautology
3. $(P(x/t) \rightarrow \neg(\forall x)\neg P)$

- Generalization
  $\langle \Gamma, \phi(x/y) \Rightarrow \Delta \rangle / \langle \Gamma, (\exists x)\phi \Rightarrow \Delta \rangle$
  $\langle \Gamma \Rightarrow \Delta, \phi(x/y) \rangle / \langle \Gamma \Rightarrow \Delta, (\forall x)\phi \rangle$
  where y is a variable substitutable for x in ϕ and y does not occur free in $\Gamma \cup \Delta \cup \{\phi\}$
- Specification
  $\langle \Gamma, \phi(x/t) \Rightarrow \Delta \rangle / \langle \Gamma, (\forall x)\phi \Rightarrow \Delta \rangle$
  $\langle \Gamma \Rightarrow \Delta, \phi(x/t) \rangle / \langle \Gamma \Rightarrow \Delta, (\exists x)\phi \rangle$
  where t is a term substitutable for x in ϕ

Example

Prove:

$$((\exists y)(\forall x)p(x,y) \to (\forall x)(\exists y)p(x,y))$$

Proof:

1. $\langle p(x,y) \Rightarrow p(x,y)\rangle$
2. $\langle (\forall x)p(x,y) \Rightarrow p(x,y)\rangle$
3. $\langle (\forall x)p(x,y) \Rightarrow (\exists y)p(x,y)\rangle$
4. $\langle (\exists y)(\forall x)p(x,y) \Rightarrow (\exists y)p(x,y)\rangle$
5. $\langle (\exists y)(\forall x)p(x,y) \Rightarrow (\forall x)(\exists y)p(x,y)\rangle$
6. $\langle \Rightarrow ((\exists y)(\forall x)p(x,y) \to (\forall x)(\exists y)p(x,y))\rangle$

Properties

A calculus is *decidable* iff for given theory T, there exists an algorithm which, given an arbitrary formula ϕ, will always says if $T \vdash \phi$ or $T \nvdash \phi$.
A calculus is *semidecidable* iff for given theory T, there exists an algorithm which, given an arbitrary formula ϕ, will always give positive answer if $T \vdash \phi$, but may give either a negative answer or no answer if $T \nvdash \phi$.

Hilbert calculus for propositional logic is sound, complete, decidable, and semidecidable.
Hilbert calculus for first-order logic is sound, complete, and semidecidable, but not decidable.
Gentzen calculus for propositional logic is sound, complete, decidable, and semidecidable.
Gentzen calculus for first-order logic is sound, complete, and semidecidable, but not decidable.

Resolution

General inference rule

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

Inference rule for (disjunctive) clauses:

$$\frac{a_1 \lor \cdots \lor a_i \lor \cdots \lor a_m, b_1 \lor \cdots \lor b_j \lor \cdots \lor b_n}{a_1 \lor \cdots \lor a_{i-1} \lor a_{i+1} \lor \cdots \lor a_m \lor b_1 \lor \cdots \lor b_{j-1} \lor b_{j+1} \lor \cdots \lor b_n}$$

where a_i is the complement of b_j

Resolution

1. $T \models \phi$ iff $T \wedge \neg\phi$ is not satisfiable
2. $T \wedge \neg\phi$ is transformed into CNF, we get a set of disjunctive clauses
3. the resolution rule is applied to all possible clauses that contain complementary literals
   - all repeated literals are removed
   - all clauses with complementary literals are discarded
4. if *empty clause* is derived, $T \wedge \neg\phi$ is not satisfiable, otherwise it is

Unification

Algorithm $UNIFY(t_1, t_2, \theta)$
Input: two terms or lists of terms t_1 and t_2, substitution θ
Output: the most general unifier of t_1 and t_2, or *failure*

1. if $t_1 = t_2$ then return θ
2. if t_1 is a variable then return $UNIFY\_VAR(t_1, t_2, \theta)$
3. if t_2 is a variable then return $UNIFY\_VAR(t_2, t_1, \theta)$
4. if t_1 and t_2 are compound terms with the same function symbol then return $UNIFY(ARGS(t_1), ARGS(t_2), \theta)$
5. if t_1 and t_2 are non-empty lists of terms then return $UNIFY(REST(t_1), REST(t_2), UNIFY(FIRST(t_1), FIRST(t_2), \theta))$
6. return *failure*

Unification

Algorithm $UNIFY\_VAR(x, t_2, \theta)$
Input: a variable x, a term t_2 and a substitution θ
Output: substitution or *failure*

1. if $\{x/t_1\} \in \theta$ then return $UNIFY(t_1, t_2, \theta)$
2. if $\{t_2/t_1\} \in \theta$ then return $UNIFY(x, t_1, \theta)$
3. if $OCCUR\_CHECK(x, t_2)$ then return *failure*
4. return $\theta\{x/t_2\}$

Example

Theory T:

$(\forall x)(gentleman(x) \rightarrow (\forall y)(lady(y) \rightarrow (\forall z)(insults(z, y) \rightarrow defeats(x, z))))$

$gentleman(jackie)$
$lady(peggy\_sue)$
$insults(billy\_boy, peggy\_sue)$

Formula ϕ:

$defeats(jackie, billy\_boy)$

Question: $T \models \phi$?

Sudoku

$$sudoku(a, b, x, y, 1) \vee \cdots \vee sudoku(a, b, x, y, 9)$$

$$\neg sudoku(a, b, x, y, n_1) \vee \neg sudoku(a, b, x, y, n_2), n_1 < n_2$$

$$\neg sudoku(a, b, x_1, y_1, n) \vee \neg sudoku(a, b, x_2, y_2, n), (x_1, y_1) < (x_2, y_2)$$

$$\neg sudoku(a, b_1, x, y_1, n) \vee \neg sudoku(a, b_2, x, y_2, n), (b_1, y_1) < (b_2, y_2)$$

$$\neg sudoku(a_1, b, x_1, y, n) \vee \neg sudoku(a_2, b, x_2, y, n), (a_1, x_1) < (a_2, x_2)$$