Logical reasoning and programming

First-order logic

Karel Chvalovský

CIIRC CTU

First-Order Logic (recap)

We have already used First-Order Logic (FOL), also called predicated logic, in Prolog. We have

- logical symbols
  - variables—an infinite (countable) set denoted Var
  - quantifier symbols $\forall$ and $\exists$
  - logical connectives
  - auxiliary symbols — parentheses, punctuation symbols ...
- non-logical symbols accompanied by their arity (the number of arguments)
  - function symbols
    - nullary functions are called constants
  - predicate (ralation) symbols
    - nullary predicate symbols are essentially propositional variables

The logical symbols are fixed, but the non-logical symbols form a language L.

Variants

Equality

Some symbols like $=$ can be either logical, or non-logical. We start with the later variant (FOL without equality), but later on we will use the former variant (FOL with equality). In FOL without equality we are allowed to interpret $=$ as needed. For example, it can be a general equivalence relation.

Many-sorted language

Sometimes it is convenient to talk about different types of objects and hence variables, function and predicate symbols can be accompanied by types. We can easily simulate finitely many sorts by introducing new predicates.

Terms

The set of all terms in a language L, denoted $Term_L$, is the smallest set satisfying

- every variable is a term in L,
- if f is an n-ary function in L and $t_1, \ldots, t_n$ are terms in L, then $f(t_1, \ldots, t_n)$ is a a term in L.

Formulae

Let p be an n-ary predicate symbol in L and $t_1, \ldots, t_n$ be terms in L, then $p(t_1, \ldots, t_n)$ is an atomic formula (or atom) in L.

The set of all formulae in a language L, denoted Fml_L, is the smallest set such that

- every atomic formula in L is a formula in L,
- if φ and ψ are formulae in L, X is a variable, then $\forall X \varphi$, $\exists X \varphi$, $(\neg\varphi)$, $(\varphi \rightarrow \psi)$, $(\varphi \wedge \psi)$, and $(\varphi \vee \psi)$ are formulae[1] in L.

A formula ψ is a subformula of φ if ψ is a substring of φ.

We usually write only parentheses that are necessary for unambiguous reading.

[1] A formula $\varphi \leftrightarrow \psi$ is a shortcut for $(\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi)$.

Free and bounded variables

We distinguish two types of variables in a formula φ

- free — not under a scope of a quantifier, denoted $FV(\varphi)$,
- bounded — under a scope of a quantifier, denoted $BV(\varphi)$.

$$FV(\varphi) = \begin{cases} \{\, X \mid X \text{ occurs in } \varphi \,\}, & \text{if } \varphi \text{ is atomic,} \\ FV(\psi), & \text{if } \varphi = \neg\psi, \\ FV(\psi) \cup FV(\chi), & \text{if } \varphi = \psi \circ \chi \text{ for } \circ \in \{\wedge, \vee, \rightarrow\}, \\ FV(\psi) \setminus \{X\}, & \text{if } \varphi = QX\psi \text{ for } Q \in \{\forall, \exists\}. \end{cases}$$

$$BV(\varphi) = \begin{cases} \emptyset, & \text{if } \varphi \text{ is atomic,} \\ BV(\psi), & \text{if } \varphi = \neg\psi, \\ BV(\psi) \cup BV(\chi), & \text{if } \varphi = \psi \circ \chi \text{ for } \circ \in \{\wedge, \vee, \rightarrow\}, \\ BV(\psi) \cup \{X\}, & \text{if } \varphi = QX\psi \text{ for } Q \in \{\forall, \exists\}. \end{cases}$$

It is possible that $FV(\varphi) \cap BV(\varphi) \neq \emptyset$.

Semantics

Interpretation

An interpretation for a language L, denoted $\mathcal{M} = (D, i)$, consists of a non-empty set D (domain) and a function i (interpretation) on D such that

- if f is an n-ary function symbol in L, then $i(f)\colon D^n \to D$,
- if p is an n-ary preditace symbol in L, then $i(p) \subseteq D^n$.

Evaluation

Let $\mathcal{M} = (D, r)$ be an interpretation for L, an evaluation in $\mathcal{M}$ is any function $e\colon \mathit{Var} \to D$. Note that $e(X \mapsto a)$, for $X \in \mathit{Var}$ and $a \in D$, is the same as e, but gives a to X.

The value of term t under an evaluation e in $\mathcal{M} = (D, i)$, denoted $t^{\mathcal{M}}[e]$, is defined recursively

- $X^{\mathcal{M}}[e] = e(X)$, if $X \in \mathit{Var}$,
- $f(t_1, \ldots, t_n)^{\mathcal{M}}[e] = i(f)(t_1^{\mathcal{M}}[e], \ldots, t_n^{\mathcal{M}}[e])$, if f is n-ary function symbol.

Tarski's definition of truth

Let $\mathcal{M} = (D, i)$ be an interpretation for L, e be an evaluation in $\mathcal{M}$, then we say that a formula φ is satisfied in $\mathcal{M}$ by e, denoted $\mathcal{M} \models \varphi[e]$, or e satisfies φ in $\mathcal{M}$ if

- $\mathcal{M} \models p(t_1, \ldots, t_n)[e]$ iff $(t_1^{\mathcal{M}}[e], \ldots, t_n^{\mathcal{M}}[e]) \in i(p)$, where p is n-ary predicate symbol in L,
- $\mathcal{M} \models (\neg\psi)[e]$ iff $\mathcal{M} \not\models \psi[e]$,
- $\mathcal{M} \models (\psi \to \chi)[e]$ iff $\mathcal{M} \not\models \psi[e]$ or $\mathcal{M} \models \chi[e]$,
- $\mathcal{M} \models (\psi \wedge \chi)[e]$ iff $\mathcal{M} \models \psi[e]$ and $\mathcal{M} \models \chi[e]$,
- $\mathcal{M} \models (\psi \vee \chi)[e]$ iff $\mathcal{M} \models \psi[e]$ or $\mathcal{M} \models \chi[e]$,
- $\mathcal{M} \models (\forall X\psi)[e]$ iff for every $a \in D$ holds $\mathcal{M} \models \psi[e(X \mapsto a)]$,
- $\mathcal{M} \models (\exists X\psi)[e]$ iff exists $a \in D$ s.t. $\mathcal{M} \models \psi[e(X \mapsto a)]$.

A formula φ is satisfiable, if there is $\mathcal{M}$ and e s.t. $\mathcal{M} \models \varphi[e]$. A set of formulae Γ is satisfiable, if there is $\mathcal{M}$ and e s.t. $\mathcal{M} \models \varphi[e]$, for every $\varphi \in \Gamma$.

Semantic consequence relation

A formula φ is valid (or holds) in $\mathcal{M}$, denoted $\mathcal{M} \models \varphi$, if φ is satisfied in $\mathcal{M}$ by any evaluation e.

A formula φ follows from (or is a consequence of) a set of formula Γ, denoted $\Gamma \models \varphi$, if and only if for any interpretation $\mathcal{M}$ and evaluation e, if for every $\psi \in \Gamma$ holds $\mathcal{M} \models \psi[e]$, then $\mathcal{M} \models \varphi[e]$. We write $\models \varphi$, if $\Gamma = \emptyset$ and say that φ is valid (or holds).

$$\Gamma \models \varphi \quad \text{iff} \quad \forall \mathcal{M} \forall e (\forall \psi \in \Gamma (\mathcal{M} \models \psi[e]) \Rightarrow \mathcal{M} \models \varphi[e])$$

Note that

$$\Gamma \models \varphi \quad \text{iff} \quad \Gamma \cup \{\neg \varphi\} \text{ is unsatisfiable}$$

We say that two formulae φ and ψ are (semantically) equivalent, denoted $\varphi \equiv \psi$, if $\{\varphi\} \models \psi$ and $\{\psi\} \models \varphi$.

Basic properties

Let φ, ψ, and χ be formulae such that X does not occur free in ψ, then

- $\neg\forall X\varphi \equiv \exists X\neg\varphi$,
- $\neg\exists X\varphi \equiv \forall X\neg\varphi$,
- $\forall X\forall Y\varphi \equiv \forall Y\forall X\varphi$,
- $\exists X\exists Y\varphi \equiv \exists X\exists Y\varphi$,
- $\forall X(\varphi \land \chi) \equiv \forall X\varphi \land \forall X\chi$,
- $\exists X(\varphi \lor \chi) \equiv \exists X\varphi \lor \exists X\chi$,
- $\exists X(\varphi \rightarrow \chi) \equiv \forall X\varphi \rightarrow \exists X\chi$,
- $(\psi \land \forall X\varphi) \equiv \forall X(\psi \land \varphi)$,
- $(\psi \land \exists X\varphi) \equiv \exists X(\psi \land \varphi)$,
- $(\psi \lor \forall X\varphi) \equiv \forall X(\psi \lor \varphi)$,
- $(\psi \lor \exists X\varphi) \equiv \exists X(\psi \lor \varphi)$,
- $(\psi \rightarrow \forall X\varphi) \equiv \forall X(\psi \rightarrow \varphi)$,
- $(\psi \rightarrow \exists X\varphi) \equiv \exists X(\psi \rightarrow \varphi)$,
- $(\forall X\varphi \rightarrow \psi) \equiv \exists X(\varphi \rightarrow \psi)$,
- $(\exists X\varphi \rightarrow \psi) \equiv \forall X(\varphi \rightarrow \psi)$.

Equivalent formulae

We can freely replace (sub)formulae by equivalent formulae. More formally

Lemma

Let ψ be a subformula of a formual φ, and χ be a formula such that $\psi \equiv \chi$. A formula φ' is obtained by replacing ψ in φ by χ. It holds that $\varphi \equiv \varphi'$.

Prenexing

We say that a formula φ is in prenex form, if $\varphi = Q_1 X_1, \ldots, Q_n X_n \psi$, where $Q_1, \ldots, Q_n$ are quantifiers and ψ is an open formula.

Lemma

For every formula φ, there exists a formula ψ in prenex normal form such that $\varphi \equiv \psi$.

Proof.

By induction on the structure of the formula φ using previous equalities and renaming of bounded variables. $\square$

Substitutions

A substitution is a function that gives terms to variables. An application of a substitution σ on a formula φ, denoted $\varphi\sigma$, is a formula φ with all free occurrences of variables replaced simultaneously by their σ images. We usually denote substitutions σ, θ, and η.

A term t is substituable into a formula φ for a variable X, if no occurrence of a variable in t becomes bounded in φ with all free occurrences of X replaced by t. This directly extends to substitutions. From now one, we assume that every substitution is substituable. However, we can always avoid this by renaming bounded variables.

Note that we usually provide only the non-identity part of a substitution.

Example

Let $\sigma = \{X \mapsto f(X,Y), Y \mapsto g(X)), Z \mapsto g(X)\}$, then $\forall Z p(X,Y,Z)\sigma = \forall Z p(f(X,Y), g(X), Z)$ and $\forall Y p(X,Y,Z)\sigma$ is not substituable, but $\forall U p(X,U,Z)\sigma = \forall U(f(X,Y), U, g(X))$.

Sentences

A term is closed, if it contains no variables. A formula φ is a sentence (or closed), if it contains no free occurrence of variables. A formula φ is open, if it contains no quantifiers.

Lemma

Let φ be a sentence, σ be a substitution, $\mathcal{M}$ be an interpretation, and e be an evaluation, then

1. $\varphi\sigma = \varphi$,
2. $\mathcal{M} \models \varphi[e]$ *iff* $\mathcal{M} \models \varphi[e']$ *for every evaluation* e',
3. $\mathcal{M} \models \varphi$ *or* $\mathcal{M} \models \neg\varphi$.

Skolem functions

It is possible to get rid of existential quantifiers by introducing Skolem functions (or Skolem constants) that behave as witnesses (or choice functions).

We know that

$$\exists X \forall Y \exists Z p(X, Y, Z) \tag{1}$$

follows from

$$\forall Y p(c, Y, f(Y)) \tag{2}$$

where c and f are fresh. Although (2) does not follow from (1), they are equisatisfiable.

Skolemization

We say that a formula is in Skolem normal form if it is in prenex normal form and it contains no existential quantifiers.

We can obtain a formula in Skolem normal form from a formula φ in prenex normal form by eliminating the first existential quantifier in

$$\varphi = \forall X_1, \ldots, \forall X_n \exists Y \psi.$$

We obtain

$$\varphi' = \forall X_1, \ldots, \forall X_n \psi\{Y \mapsto f(X_1, \ldots, X_n)\}$$

where f is a fresh function. Then we repeat the whole process with φ' until there is no existential quantifier in the formula. The resulting formula is equisatisfiable with φ.

We want Skolem functions with small arity.

Usual transformations

NNF (negation normal form)

Apply the following rewriting steps as long as possible:

$$
\begin{array}{rcl}
\neg\neg\varphi & \rightsquigarrow & \varphi \\
\varphi \to \psi & \rightsquigarrow & \neg\varphi \vee \psi \\
\neg(\varphi \wedge \psi) & \rightsquigarrow & \neg\varphi \vee \neg\psi \\
\neg(\varphi \vee \psi) & \rightsquigarrow & \neg\varphi \wedge \neg\psi \\
\neg(\forall X \varphi) & \rightsquigarrow & \exists X \neg\varphi \\
\neg(\exists X \varphi) & \rightsquigarrow & \forall X \neg\varphi
\end{array}
$$

Rectified formulae

A formula φ is rectified if

- no variable occurs both free and bounded in φ,
- no two quantifiers in φ quantify over the same variable.

We obtain a rectified formula by renaming bounded variables.

Clauses in FOL

We adapt our terminology from propositional logic.

A literal l is an atomic formula (positive), or a negation of an atomic formula (negative).

A clause is a disjunction of finitely many literals. An important special case is the empty clause, denoted $\Box$.

A formula φ is in conjunctive normal form (CNF) if φ is a conjunction of clauses.

Recall two special cases. The empty clause $\Box$ (empty disjunction) is unsatisfiable. The empty conjunction is satisfiable.

CNF

The universal closure of a formula φ, denoted $\forall\varphi$, is a formula $\forall X_1, \ldots \forall X_n \varphi$, where $X_1, \ldots, X_n$ are all free variables in φ.

We produce a CNF φ' (implicitly universally quantified) from a sentence φ by performing the following steps

1. produce a NNF,
2. rectify,
3. skolemize (an obvious generalization for sentences not in prenex form),
4. remove all universal quantifiers,
5. produce a CNF as in propositional logic.

Let φ' be a set of clauses $\chi_1, \ldots, \chi_n$. It holds that

$$\varphi \text{ is satisf.} \quad \text{iff} \quad \forall \bigwedge \varphi' \text{ is satisf.} \quad \text{iff} \quad \forall \chi_1 \wedge \cdots \wedge \forall \chi_n \text{ is satisf.}$$

where $\forall \bigwedge \varphi'$ is $\forall(\chi_1 \wedge \cdots \wedge \chi_n)$.

Our problem

Let $\Gamma = \{\psi_1, \ldots, \psi_n\}$ be a set of senteces and φ be a sentence. We know that

$$\Gamma \models \varphi$$

iff

$$\Gamma \cup \{\neg\varphi\} \text{ is unsatisfiable}$$

iff

$$\forall\psi_1' \cup \cdots \cup \forall\psi_n' \cup \forall(\neg\varphi)' \text{ is unsatisfiable,}$$

where $\psi_1', \ldots, \psi_n', (\neg\varphi)'$ are $\psi_1, \ldots, \psi_n, \neg\varphi$ in CNF (=clauses) and $\forall\Delta = \{\, \forall\chi \mid \chi \in \Delta \,\}$ for a set of clauses Δ.

We say that Δ is a set of clauses assuming that it is implicitly universally quantified.

Bibliography I

Robinson, John Alan and Andrei Voronkov, eds. (2001). *Handbook of Automated Reasoning*. Vol. 1. Elsevier Science.