First-Order Logic

Syntax

Definition

The *alphabet of a first-order language* is organised into the following categories.

- *Logical connectives*: $\bot$, $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\forall$ and $\exists$.
- *Auxiliary symbols*: "." , "," , "(" and ")".
- *Variables*: we assume a countable infinite set $\mathcal{X}$ of variables, ranged over by $x, y, z, \ldots$.
- *Constant symbols*: we assume a countable set $\mathcal{C}$ of constant symbols, ranged over by $a, b, c, \ldots$.
- *Function symbols*: we assume a countable set $\mathcal{F}$ of function symbols, ranged over by $f, g, h, \ldots$. Each function symbol f has a fixed arity $\mathsf{ar}(f)$, which is a positive integer.
- *Predicate symbols*: we assume a countable set $\mathcal{P}$ of predicate symbols, ranged over by $P, Q, R, \ldots$. Each predicate symbol P has a fixed arity $\mathsf{ar}(P)$, which is a non-negative integer. (Predicate symbols with arity 0 play the role of propositions.)

The union of the non-logical symbols of the language is called the *vocabulary* and is denoted by $\mathcal{V}$, i.e. $\mathcal{V} = \mathcal{C} \cup \mathcal{F} \cup \mathcal{P}$.

Notation

Throughout, and when not otherwise said, we assume a vocabulary $\mathcal{V} = \mathcal{C} \cup \mathcal{F} \cup \mathcal{P}$.

Syntax

Definition

The set of *terms* of a first-order language over a vocabulary $\mathcal{V}$ is given by:

$$\mathbf{Term}_{\mathcal{V}} \ni t, u ::= x \mid c \mid f(t_1, \ldots, t_{\mathrm{ar}(f)})$$

The set of *variables occurring* in t is denoted by $\mathrm{Vars}(t)$.

Definition

The set of *formulas* of a first-order language over a vocabulary $\mathcal{V}$ is given by:

$$\begin{aligned}\mathbf{Form}_{\mathcal{V}} \ni \phi, \psi, \theta \quad ::= \quad & P(t_1, \ldots, t_{\mathrm{ar}(P)}) \mid \bot \mid (\neg \phi) \mid (\phi \wedge \psi) \mid (\phi \vee \psi) \\ & \mid (\phi \rightarrow \psi) \mid (\forall x . \phi) \mid (\exists x . \phi)\end{aligned}$$

An *atomic formula* has the form $\bot$ or $P(t_1, \ldots, t_{\mathrm{ar}(P)})$.

Remark

- *We assume the conventions of propositional logic to omit parentheses, and additionally assume that quantifiers have the lowest precedence.*
- *Nested quantifications such as* $\forall x . \forall y . \phi$ *are abbreviated to* $\forall x, y . \phi$.
- *There are recursion and induction principles (e.g. structural ones) for* $\mathbf{Term}_{\mathcal{V}}$ *and* $\mathbf{Form}_{\mathcal{V}}$.

Syntax

Definition

- A formula ψ that occurs in a formula ϕ is called a *subformula* of ϕ.
- In a quantified formula $\forall x \,.\, \phi$ or $\exists x \,.\, \phi$, x is the *quantified variable* and ϕ is the *scope* of the quantification.
- Occurrences of the quantified variable within the respective scope are said to be *bound*. Variable occurrences that are not bound are said to be *free*.
- The set of *free variables* (resp.*bound variables*) of a formula θ, is denoted $\mathrm{FV}(\theta)$ (resp. $\mathrm{BV}(\theta)$).

Definition

- A *sentence* (or *closed* formula) is a formula without free variables.
- If $\mathrm{FV}(\phi) = \{x_1, \ldots, x_n\}$, the *universal closure* of ϕ is the formula $\forall x_1, \ldots, x_n \,.\, \phi$ and the *existential closure* of ϕ is the formula $\exists x_1, \ldots, x_n \,.\, \phi$.

Definition

- A *substitution* is a mapping $\sigma : \mathcal{X} -> \mathbf{Term}_{\mathcal{V}}$ s.t. the set $\mathrm{dom}(\sigma) = \{x \in \mathcal{X} \mid \sigma(x) \neq x\}$, called the *substitution domain*, is finite.
- The notation $[t_1/x_1, \ldots, t_n/x_n]$ (for distinct x_i's) denotes the substitution whose domain is contained in $\{x_1, \ldots, x_n\}$ and maps each x_i to t_i.

Syntax

Definition

The *application of a substitution* σ *to a term* t is denoted by $t\,\sigma$ and is defined recursively by:

$$\begin{array}{rcl} x\,\sigma & = & \sigma(x) \\ c\,\sigma & = & c \\ f(t_1,\ldots,t_{\mathrm{ar}(f)})\,\sigma & = & f(t_1\,\sigma,\ldots,t_{\mathrm{ar}(f)}\,\sigma) \end{array}$$

Remark

The result of

$$t\,[t_1/x_1,\ldots,t_n/x_n]$$

corresponds to the simultaneous substitution of $t_1,\ldots,t_n$ *for* $x_1,\ldots,x_n$ *in* t*. This differs from the application of the corresponding singleton substitutions in sequence,*

$$((t\ [t_1/x_1])\ldots)\,[t_n/x_n]\,.$$

Notation

Given a function $f : X \longrightarrow Y$*,* $x \in X$ *and* $y \in Y$*, the notation* $f[x \mapsto y]$ *stands for the function defined as* f *except possibly for* x*, to which* y *is assigned, called the patching of* f *in* x *to* y*.*

Syntax

Definition

The *application of a substitution* σ *to a formula* ϕ, written $\phi\,\sigma$, is given recursively by:

$$
\begin{aligned}
\bot\,\sigma &= \bot \\
P(t_1,\ldots,t_{\mathrm{ar}(P)})\,\sigma &= P(t_1\,\sigma,\ldots,t_{\mathrm{ar}(P)}\,\sigma) \\
(\neg\phi)\,\sigma &= \neg(\phi\,\sigma) \\
(\phi\odot\psi)\,\sigma &= (\phi\,\sigma)\odot(\psi\,\sigma) \\
(Qx\,.\,\phi)\,\sigma &= Qx\,.\,(\phi(\sigma[x\mapsto x]))
\end{aligned}
$$

where $\odot\in\{\wedge,\vee\ \rightarrow\}$ and $Q\in\{\forall,\exists\}$.

Remark

- *Only free occurrences of variables can change when a substitution is applied to a formula.*
- *Unrestricted application of substitutions to formulas can cause capturing of variables as in:* $(\forall x.\,P(x,y))\,[g(x)/y]=\forall x.\,P(x,g(x))$
- *"Safe substitution" (which we assume throughout) is achieved by imposing that a substitution when applied to a formula should be free for it.*

Definition

- A *term* t *is free for* x *in* θ iff x has no free occurrences in the scope of a quantifier Qy ($y\neq x$) s.t. $y\in\mathrm{Vars}(t)$.
- A *substitution* σ *is free for* θ iff $\sigma(x)$ is free for x in θ, for all $x\in\mathrm{dom}(\sigma)$.

Semantics

Definition

Given a vocabulary $\mathcal{V}$, a $\mathcal{V}$ *-structure* is a pair $\mathcal{M} = (D, I)$ where D is a nonempty set, called the *interpretation domain*, and I is called the *interpretation function*, and assigns constants, functions and predicates over D to the symbols of $\mathcal{V}$ as follows:

- for each $c \in \mathcal{C}$, the interpretation of c is a constant $I(c) \in D$;
- for each $f \in \mathcal{F}$, the interpretation of f is a function $I(f) : D^{\mathrm{ar}(f)} \longrightarrow D$;
- for each $P \in \mathcal{P}$, the interpretation of P is a function $I(P) : D^{\mathrm{ar}(P)} \longrightarrow \{\mathbf{F}, \mathbf{T}\}$. In particular, 0-ary predicate symbols are interpreted as truth values.

$\mathcal{V}$ -structures are also called *models* for $\mathcal{V}$.

Definition

Let D be the interpretation domain of a structure. An *assignment* for D is a function $\alpha : \mathcal{X} \longrightarrow D$ from the set of variables to the domain D.

Notation

In what follows, we let $\mathcal{M}, \mathcal{M}', \ldots$ range over the structures of an intended vocabulary, and $\alpha, \alpha', \ldots$ range over the assignments for the interpretation domain of an intended structure.

Semantics

Definition

Let $\mathcal{M} = (D, I)$ be a $\mathcal{V}$ -structure and α an assignment for D.

- The *value of a term t w.r.t. $\mathcal{M}$ and α* is an element of D, denoted by $[\![t]\!]_{\mathcal{M},\alpha}$, and recursively given by:

$$
\begin{array}{lcl}
[\![x]\!]_{\mathcal{M},\alpha} & = & \alpha(x) \\
[\![c]\!]_{\mathcal{M},\alpha} & = & I(c) \\
[\![f(t_1,\ldots,t_{\mathrm{ar}(f)})]\!]_{\mathcal{M},\alpha} & = & I(f)([\![t_1]\!]_{\mathcal{M},\alpha},\ldots,[\![t_{\mathrm{ar}(f)}]\!]_{\mathcal{M},\alpha})
\end{array}
$$

- The *(truth) value of a formula ϕ w.r.t. $\mathcal{M}$ and α*, is denoted by $[\![\phi]\!]_{\mathcal{M},\alpha}$, and recursively given by:

$$
\begin{array}{lcl}
[\![\bot]\!]_{\mathcal{M},\alpha} = \mathbf{F} & & \\
[\![P(t_1,\ldots,t_{\mathrm{ar}(P)})]\!]_{\mathcal{M},\alpha} & = & I(P)([\![t_1]\!]_{\mathcal{M},\alpha},\ldots,[\![t_{\mathrm{ar}(P)}]\!]_{\mathcal{M},\alpha}) \\
[\![\neg\phi]\!]_{\mathcal{M},\alpha} = \mathbf{T} & \text{iff} & [\![\phi]\!]_{\mathcal{M},\alpha} = \mathbf{F} \\
[\![\phi \wedge \psi]\!]_{\mathcal{M},\alpha} = \mathbf{T} & \text{iff} & [\![\phi]\!]_{\mathcal{M},\alpha} = \mathbf{T} \text{ and } [\![\psi]\!]_{\mathcal{M},\alpha} = \mathbf{T} \\
[\![\phi \vee \psi]\!]_{\mathcal{M},\alpha} = \mathbf{T} & \text{iff} & [\![\phi]\!]_{\mathcal{M},\alpha} = \mathbf{T} \text{ or } [\![\psi]\!]_{\mathcal{M},\alpha} = \mathbf{T} \\
[\![\phi \to \psi]\!]_{\mathcal{M},\alpha} = \mathbf{T} & \text{iff} & [\![\phi]\!]_{\mathcal{M},\alpha} = \mathbf{F} \text{ or } [\![\psi]\!]_{\mathcal{M},\alpha} = \mathbf{T} \\
[\![\forall x.\,\phi]\!]_{\mathcal{M},\alpha} = \mathbf{T} & \text{iff} & [\![\phi]\!]_{\mathcal{M},\alpha[x\mapsto a]} = \mathbf{T} \text{ for all } a \in D \\
[\![\exists x.\,\phi]\!]_{\mathcal{M},\alpha} = \mathbf{T} & \text{iff} & [\![\phi]\!]_{\mathcal{M},\alpha[x\mapsto a]} = \mathbf{T} \text{ for some } a \in D
\end{array}
$$

Semantics

Remark

Universal and existential quantifications are indeed a gain over PL. They can be read (resp.) as generalised conjunction and disjunction (possibly infinite):

$$[\![\forall x.\,\phi]\!]_{\mathcal{M},\alpha} = \bigwedge_{a\in D} [\![\phi]\!]_{\mathcal{M},\alpha[x\mapsto a]} \qquad [\![\exists x.\,\phi]\!]_{\mathcal{M},\alpha} = \bigvee_{a\in D} [\![\phi]\!]_{\mathcal{M},\alpha[x\mapsto a]}$$

Definition

Let $\mathcal{V}$ be a vocabulary and $\mathcal{M}$ a $\mathcal{V}$-structure.

- $\mathcal{M}$ *satisfies* ϕ *with* α, denoted by $\mathcal{M}, \alpha \models \phi$, iff $[\![\phi]\!]_{\mathcal{M},\alpha} = \mathbf{T}$.
- $\mathcal{M}$ *satisfies* ϕ (or that ϕ *is valid in* $\mathcal{M}$, or $\mathcal{M}$ is a *model* of ϕ), denoted by $\mathcal{M} \models \phi$, iff for every assignment α, $\mathcal{M}, \alpha \models \phi$.
- ϕ is *satisfiable* if exists $\mathcal{M}$ s.t. $\mathcal{M} \models \phi$, and it is *valid*, denoted by $\models \phi$, if $\mathcal{M} \models \phi$ for every $\mathcal{M}$. ϕ is *unsatisfiable* (or a *contradiction*) if it is not satisfiable, and refutable if it is not valid.

Lemma

Let $\mathcal{M}$ be a structure, t and u terms, ϕ a formula, and α, α' assignments.

- *If for all* $x \in \mathrm{Vars}(t)$, $\alpha(x) = \alpha'(x)$, *then* $[\![t]\!]_{\mathcal{M},\alpha} = [\![t]\!]_{\mathcal{M},\alpha'}$
- *If for all* $x \in \mathrm{FV}(\phi)$, $\alpha(x) = \alpha'(x)$, *then* $\mathcal{M}, \alpha \models \phi$ *iff* $\mathcal{M}, \alpha' \models \phi$.
- $[\![t\,[u/x]]\!]_{\mathcal{M},\alpha} = [\![t]\!]_{\mathcal{M},\alpha[x\mapsto [\![u]\!]_{\mathcal{M},\alpha}]}$
- *If t is free for x in ϕ, then* $\mathcal{M}, \alpha \models \phi\,[t/x]$ *iff* $\mathcal{M}, \alpha[x \mapsto [\![\mathcal{M}]\!]_{\alpha,t}] \models \phi$.

Semantics

Proposition (Lifting validity of PL)

*Let $\lceil \cdot \rceil$: **Prop** $->$ **Form**$_{\mathcal{V}}$, be a mapping from the set of proposition symbols to first-order formulas and denote also by $\lceil \cdot \rceil$ its homomorphic extension to all propositional formulas. Then, for all propositional formulas A and B:*

- $\mathcal{M}, \alpha \models \lceil A \rceil$ *iff* $\overline{\mathcal{M}_\alpha} \models_{PL} A$, *where* $\overline{\mathcal{M}_\alpha} = \{P \mid \mathcal{M}, \alpha \models \lceil P \rceil\}$.
- *If* $\models_{PL} A$, *then* $\models_{FOL} \lceil A \rceil$.
- *If* $A \equiv_{PL} B$, *then* $\lceil A \rceil \equiv_{FOL} \lceil B \rceil$.

Some properties of logical equivalence

- The properties of logical equivalence listed for PL hold for FOL.
- The following equivalences hold:

$$\neg \forall x . \phi \equiv \exists x . \neg \phi \qquad\qquad \neg \exists x . \phi \equiv \forall x . \neg \phi$$

$$\forall x . \phi \wedge \psi \equiv (\forall x . \phi) \wedge (\forall x . \psi) \qquad\qquad \exists x . \phi \vee \psi \equiv (\exists x . \phi) \vee (\exists x . \psi)$$

- For $Q \in \{\forall, \exists\}$, if y is free for x in ϕ and $y \notin \mathrm{FV}(\phi)$, then $Qx \,.\, \phi \equiv Qy \,.\, \phi\,[y/x]$.
- For $Q \in \{\forall, \exists\}$, if $x \notin \mathrm{FV}(\phi)$, then $Qx \,.\, \phi \equiv \phi$.
- For $Q \in \{\forall, \exists\}$ and $\odot \in \{\wedge, \vee\}$, if $x \notin \mathrm{FV}(\psi)$, then $Qx \,.\, \phi \odot \psi \equiv (Qx \,.\, \phi) \odot \psi$.

Semantics

Definition

A formula is in *prenex form* if it is of the form $Q_1 x_1 . Q_2 x_2 . \dots Q_n x_n . \psi$ (possibly with $n = 0$) where each Q_i is a quantifier (either $\forall$ or $\exists$) and ψ is a quantifier-free formula .

Proposition

For any formula of first-order logic, there exists an equivalent formula in prenex form.

Proof.

Such a prenex form can be obtained by rewriting, using the logical equivalences listed before. □

Remark

Unlike PL, the validity problem of FOL is not decidable, but it is semi-decidable, i.e. there are procedures s.t., given a formula ϕ, they terminate with "yes" if ϕ is valid but may fail to terminate if ϕ is not valid.

Semantics

Definition

- $\mathcal{M}$ satisfies Γ with α, denoted by $\mathcal{M}, \alpha \models \Gamma$, if $\mathcal{M}, \alpha \models \phi$ for every $\phi \in \Gamma$.
- The notions of satisfiable, valid, unsatisfiable and refutable set of formulas are defined in the expected way.
- Γ *entails* ϕ (or ϕ is a *logical consequence* of Γ), denoted by $\Gamma \models \phi$, iff for every structure $\mathcal{M}$ and assignment α, if $\mathcal{M}, \alpha \models \Gamma$ then $\mathcal{M}, \alpha \models \phi$.
- ϕ is *logically equivalent* to ψ , denoted by $\phi \equiv \psi$, iff $[\![\phi]\!]_{\mathcal{M},\alpha} = [\![\psi]\!]_{\mathcal{M},\alpha}$ for every structure $\mathcal{M}$ and assignment α.

Some properties of semantic entailment

- The properties of semantic entailment listed for PL hold for FOL.
- If t is free for x in ϕ and $\Gamma \models \forall x.\phi$, then $\Gamma \models \phi[t/x]$.
- If $x \notin \mathsf{FV}(\Gamma)$ and $\Gamma \models \phi$, then $\Gamma \models \forall x.\phi$.
- If t is free for x in ϕ and $\Gamma \models \phi[t/x]$, then $\Gamma \models \exists x.\phi$.
- If $x \notin \mathsf{FV}(\Gamma \cup \{\psi\})$, $\Gamma \models \exists x.\phi$ and $\Gamma, \phi \models \psi$, then $\Gamma \models \psi$.

Proof system

The natural deduction system $\mathcal{N}_{\mathsf{FOL}}$

- The proof system for FOL we consider is a natural deduction system in sequent style extending $\mathcal{N}_{\mathsf{PL}}$.
- The various definitions made in the context of $\mathcal{N}_{\mathsf{PL}}$ carry over to $\mathcal{N}_{\mathsf{FOL}}$. The difference is that$\mathcal{N}_{\mathsf{FOL}}$ deals with first-order formulas and it has additional introduction and elimination rules to deal with the quantifiers.

Quantifier rules of $\mathcal{N}_{\mathsf{FOL}}$

$$(\mathsf{I}_\forall)\ \frac{\Gamma \vdash \phi\,[y/x]}{\Gamma \vdash \forall x.\,\phi}\ (\mathrm{a}) \qquad\qquad (\mathsf{E}_\forall)\ \frac{\Gamma \vdash \forall x.\,\phi}{\Gamma \vdash \phi\,[t/x]}$$

$$(\mathsf{I}_\exists)\ \frac{\Gamma \vdash \phi\,[t/x]}{\Gamma \vdash \exists x.\,\phi} \qquad\qquad (\mathsf{E}_\exists)\ \frac{\Gamma \vdash \exists x.\,\phi \qquad \Gamma, \phi\,[y/x] \vdash \theta}{\Gamma \vdash \theta}\ (\mathrm{b})$$

(a) $y \notin \mathsf{FV}(\Gamma)$ and either $x = y$ or $y \notin \mathsf{FV}(\phi)$.
(b) $y \notin \mathsf{FV}(\Gamma \cup \{\theta\})$ and either $x = y$ or $y \notin \mathsf{FV}(\phi)$.
(c) Recall that we assume safe substitution, i.e. in a substitution $\phi[t/x]$, we assume that t is free for x in ϕ.

Remark

The properties of $\mathcal{N}_{\mathsf{PL}}$ can be extended to $\mathcal{N}_{\mathsf{FOL}}$, in particular the soundness and completeness theorems.

Theorem (Adequacy)

$\Gamma \models \varphi$ *iff* $\Gamma \vdash \varphi$.

First-order theories

Definition

Let $\mathcal{V}$ be a vocabulary of a first-order language.

- A first-order *theory* $\mathcal{T}$ is a set of $\mathcal{V}$ -sentences that is closed under derivability (i.e., $\mathcal{T} \vdash \phi$ implies $\phi \in \mathcal{T}$). A $\mathcal{T}$*-structure* is a $\mathcal{V}$ -structure that validates every formula of $\mathcal{T}$.
- A formula ϕ is $\mathcal{T}$*-valid* (resp. $\mathcal{T}$*-satisfiable*) if every (resp. some) $\mathcal{T}$-structure validates ϕ. $\mathcal{T} \models \phi$ denotes the fact that ϕ is $\mathcal{T}$-valid.
- Other concepts regarding validity of first-order formulas are carried over to theories in the obvious way.

Definition

A subset $\mathcal{A} \subseteq \mathcal{T}$ is called an *axiom set* for the theory $\mathcal{T}$ when $\mathcal{T}$ is the deductive closure of $\mathcal{A}$, i.e. $\psi \in \mathcal{T}$ iff $\mathcal{A} \vdash \psi$, or equivalently, iff $\vdash \psi$ can be derived in $\mathcal{N}_{\mathsf{FOL}}$ with an axiom-schema:

$$\frac{}{\Gamma \vdash \phi} \text{ if } \phi \in \mathcal{A}$$

.

First-order theories

Equality theory

The *theory of equality* $\mathcal{T}_E$ for $\mathcal{V}$ (which is assumed to have a binary equality predicate symbol "$=$") has the following axiom set:

- *reflexivity:* $\forall x \,.\, x = x$
- *symmetry:* $\forall x, y \,.\, x = y \rightarrow y = x$
- *transitivity:* $\forall x, y, z \,.\, x = y \wedge y = z \rightarrow x = z$
- *congruence for function symbols:* for every $f \in \mathcal{F}$ with $\mathrm{ar}(f) = n$,

$$\forall \overline{x}, \overline{y} \,.\, x_1 = y_1 \wedge \ldots \wedge x_n = y_n \rightarrow f(x_1, \ldots, x_n) = f(y_1, \ldots, y_n)$$

- *congruence for predicate symbols:* for every $P \in \mathcal{P}$ with $\mathrm{ar}(P) = n$,

$$\forall \overline{x}, \overline{y} \,.\, x_1 = y_1 \wedge \ldots \wedge x_n = y_n \rightarrow P(x_1, \ldots, x_n) \rightarrow P(y_1, \ldots, y_n)$$

Theorem

A sentence ϕ is valid in all normal structures (i.e. structures which interpret $=$ as the equality relation over the interpretation domain) iff $\phi \in \mathcal{T}_E$.