

CAS 734 Winter 2014

03 Review of First-Order Logic

William M. Farmer

Department of Computing and Software
McMaster University

18 January 2014



What is First-Order Logic?

- First-order logic is the study of statements about individuals using functions, predicates, and quantification.
 - ▶ First-order logic is also called first-order predicate logic and first-order quantificational logic.
- First-order logic is propositional logic plus:
 - ▶ Terms that denote individuals.
 - ▶ Predicates that are applied to terms.
 - ▶ Quantifiers applied to individual variables.
- First-order logic is “first-order” because quantification is over individuals but not over higher-order objects such as functions and predicates.
- There are many versions of first-order logic.
- We will define and employ a version of first-order logic named FOL.

Syntax of FOL: Languages

- Let \mathcal{V} be a fixed infinite set of symbols called **variables**.
- A **language** of FOL is a triple $L = (\mathcal{C}, \mathcal{F}, \mathcal{P})$ where:
 - ▶ \mathcal{C} is a set of symbols called **individual constants**.
 - ▶ \mathcal{F} is a set of symbols called **function symbols**, each with an assigned arity ≥ 1 .
 - ▶ \mathcal{P} is a set of symbols called **predicate symbols**, each with an assigned arity ≥ 1 . \mathcal{P} contains the binary predicate symbol $=$.
 - ▶ \mathcal{V} , \mathcal{C} , \mathcal{F} , and \mathcal{P} are pairwise disjoint.

Syntax of FOL: Terms and Formulas

- Let $L = (\mathcal{C}, \mathcal{F}, \mathcal{P})$ be a language of FOL.
- A **term** of L is a string of symbols inductively defined by the following formation rules:
 - ▶ Each $x \in \mathcal{V}$ and $a \in \mathcal{C}$ is a term of L .
 - ▶ If $f \in \mathcal{F}$ is n -ary and t_1, \dots, t_n are terms of L , then $f(t_1, \dots, t_n)$ is a term of L .
- A **formula** of L is a string of symbols inductively defined by the following formation rules:
 - ▶ If $p \in \mathcal{P}$ is n -ary and t_1, \dots, t_n are terms of L , then $p(t_1, \dots, t_n)$ is a formula of L .
 - ▶ If A and B are formulas of L and $x \in \mathcal{V}$, then $(\neg A)$ and $(A \Rightarrow B)$, and $(\forall x . A)$ are formulas of L .
- $=$, \neg , \Rightarrow , and \forall are the **logical constants** of FOL.

Syntax of FOL: Notational Definitions

$(s = t)$	denotes	$= (s, t).$
$(s \neq t)$	denotes	$(\neg(s = t)).$
T	denotes	$(\forall x . (x = x)).$
F	denotes	$(\neg(T)).$
$(A \vee B)$	denotes	$((\neg A) \Rightarrow B).$
$(A \wedge B)$	denotes	$(\neg((\neg A) \vee (\neg B))).$
$(A \Leftrightarrow B)$	denotes	$((A \Rightarrow B) \wedge (B \Rightarrow A)).$
$(\exists x . A)$	denotes	$(\neg(\forall x . (\neg A))).$
$(\Box x_1, \dots, x_n . A)$	denotes	$(\Box x_1 . (\Box x_2, \dots, x_n . A))$ where $n \geq 2$ and $\Box \in \{\forall, \exists\}.$

Free and Bound Variables

- The **scope** of a quantifier $\forall x$ or $\exists x$ in a formula $\forall x . B$ or $\exists x . B$, respectively, is the part of B that is not in a subformula of B of the form $\forall x . C$ or $\exists x . C$.
- An occurrence of a variable x in a formula A is **free** if it is not in the scope of a quantifier $\forall x$ or $\exists x$; otherwise the occurrence of x in A is **bound**.
 - ▶ An occurrence of a variable in a formula is either free or bound but never both.
 - ▶ A variable can be both bound and free in a formula.
- A formula is **closed** if it contains no free variables.
- A **sentence** is a closed formula.

Substitution

- Let x be a variable, t a term, and A a formula.
- The **substitution** of t for x in A , written

$$A[x \mapsto t] \text{ or } A[t/x],$$

is the result of replacing each free occurrence of x in A with t .

- Suppose A is $\forall y . x = y$ and t is $f(y)$. Then the substitution $A[x \mapsto t]$ is said to **capture** y .
 - ▶ Variable captures often produce unsound results.
- **t is free for x in A** if no free occurrence of x in A is in the scope of $\forall y$ or $\exists y$ for any variable y occurring t .
 - ▶ Hence, t is free for x in A if the substitution $A[x \mapsto t]$ does not result in any variable captures.

Semantics of FOL: Models

- A **model** for a language $L = (\mathcal{C}, \mathcal{F}, \mathcal{P})$ of FOL is a pair $M = (D, I)$ where D is a nonempty domain (set) and I is a total function on $\mathcal{C} \cup \mathcal{F} \cup \mathcal{P}$ such that:
 - ▶ If $a \in \mathcal{C}$, $I(a) \in D$.
 - ▶ If $f \in \mathcal{F}$ is n -ary, $I(f) : D^n \rightarrow D$ and $I(f)$ is total.
 - ▶ If $p \in \mathcal{P}$ is n -ary, $I(p) : D^n \rightarrow \{\text{T}, \text{F}\}$ and $I(p)$ is total.
 - ▶ $I(=)$ is id_D , the identity predicate on D .
- A **variable assignment** into M is a function that maps each $x \in \mathcal{V}$ to an element of D .
- Given a variable assignment φ into M , $x \in \mathcal{V}$, and $d \in D$, let $\varphi[x \mapsto d]$ be the variable assignment φ' into M such $\varphi'(x) = d$ and $\varphi'(y) = \varphi(y)$ for all $y \neq x$.

Semantics of FOL: Valuation Function

The **valuation function** for a model M for a language $L = (\mathcal{C}, \mathcal{F}, \mathcal{P})$ of FOL is the binary function V^M that satisfies the following conditions for all variable assignments φ into M and all terms t and formulas A of L :

1. Let $t \in \mathcal{V}$. Then $V_{\varphi}^M(t) = \varphi(t)$.
2. Let $t \in \mathcal{C}$. Then $V_{\varphi}^M(t) = I(t)$.
3. Let $t = f(t_1, \dots, t_n)$. Then $V_{\varphi}^M(t) = I(f)(V_{\varphi}^M(t_1), \dots, V_{\varphi}^M(t_n))$.
4. Let $A = p(t_1, \dots, t_n)$. Then $V_{\varphi}^M(A) = I(p)(V_{\varphi}^M(t_1), \dots, V_{\varphi}^M(t_n))$.
5. Let $A = (\neg A')$. If $V_{\varphi}^M(A') = \text{F}$, then $V_{\varphi}^M(A) = \text{T}$; otherwise $V_{\varphi}^M(A) = \text{F}$.
6. Let $A = (A_1 \Rightarrow A_2)$. If $V_{\varphi}^M(A_1) = \text{T}$ and $V_{\varphi}^M(A_2) = \text{F}$, then $V_{\varphi}^M(A) = \text{F}$; otherwise $V_{\varphi}^M(A) = \text{T}$.
7. Let $A = (\forall x . A')$. If $V_{\varphi[x \mapsto d]}^M(A') = \text{T}$ for all $d \in D$, then $V_{\varphi}^M(A) = \text{T}$; otherwise $V_{\varphi}^M(A) = \text{F}$.

Notes on Quantifiers

- The universal and existential quantifiers are duals of each other:

$$\neg(\forall x . A) \Leftrightarrow \exists x . \neg A, \quad \neg(\exists x . A) \Leftrightarrow \forall x . \neg A.$$

- Changing the order of quantifiers in a formula usually changes the meaning of the formula.
 - ▶ As a rule, $\forall x . \exists y . A \not\Leftrightarrow \exists y . \forall x . A$.
- In a formula of the form $\forall x . \exists y . A$, the value of the existentially quantified variable y depends on the value of the universally quantified variable x .
- A universal statement like “All rodents are mammals” is formalized as $\forall x . \text{rodent}(x) \Rightarrow \text{mammal}(x)$.
- An existential statement like “Some mammals are rodents” is formalized as $\exists x . \text{mammal}(x) \wedge \text{rodent}(x)$.

Algebras as Models

- If $L = (\mathcal{C}, \mathcal{F}, \mathcal{P})$ is a finite language of FOL, we may present the language as

$$L = (c_1, \dots, c_k, f_1, \dots, f_m, p_1, \dots, p_n)$$

where $\mathcal{C} = \{c_1, \dots, c_k\}$, $\mathcal{F} = \{f_1, \dots, f_m\}$, and $\mathcal{P} = \{p_1, \dots, p_n\}$.

- An algebra

$$(D, d_1, \dots, d_k, g_1, \dots, g_m, q_1, \dots, q_n)$$

can then be considered a model for L if $M = (D, I)$ is a model for L where:

1. $I(c_i) = d_i$ for $1 \leq i \leq k$.
2. $I(f_i) = g_i$ for $1 \leq i \leq m$.
3. $I(p_i) = q_i$ for $1 \leq i \leq n$.

Metatheorems of FOL

- **Completeness Theorem (Gödel 1930)**. There is a sound and complete proof system for FOL.
- **Compactness Theorem**. Let Σ be a set of formulas of a language of FOL. If Σ is finitely satisfiable, then Σ is satisfiable.
- **Undecidability Theorem (Church 1936)**. First-order logic is undecidable. That is, for some language L of FOL, the problem of whether or not a given formula of L is valid is undecidable.

A Hilbert-Style Proof System (1/2)

Let **H** be the following Hilbert-style proof system for a language L of FOL:

- The **logical axioms** of **H** are all formulas of L that are instances of the following schemas:
 - ▶ For propositional logic:
 - A1: $A \Rightarrow (B \Rightarrow A)$.
 - A2: $(A \Rightarrow (B \Rightarrow C)) \Rightarrow ((A \Rightarrow B) \Rightarrow (A \Rightarrow C))$.
 - A3: $(\neg A \Rightarrow \neg B) \Rightarrow (B \Rightarrow A)$.
 - ▶ For quantification:
 - A4: $(\forall x . (A \Rightarrow B)) \Rightarrow (A \Rightarrow (\forall x . B))$
provided x is not free in A .
 - A5: $(\forall x . A) \Rightarrow A[x \mapsto t]$
provided t is free for x in A .

A Hilbert-Style Proof System (2/2)

- For equality:

A6: $\forall x . x = x.$

A7: $\forall x, y . x = y \Rightarrow y = x.$

A8: $\forall x, y, z . (x = y \wedge y = z) \Rightarrow x = z.$

A9: $\forall x_1, \dots, x_n, y_1, \dots, y_n .$
 $(x_1 = y_1 \wedge \dots \wedge x_n = y_n) \Rightarrow$
 $f(x_1, \dots, x_n) = f(y_1, \dots, y_n)$
where $f \in \mathcal{F}$ is n -ary.

A10: $\forall x_1, \dots, x_n, y_1, \dots, y_n .$
 $(x_1 = y_1 \wedge \dots \wedge x_n = y_n) \Rightarrow$
 $(p(x_1, \dots, x_n) \Leftrightarrow p(y_1, \dots, y_n))$
where $p \in \mathcal{P}$ is n -ary.

- The rules of inference of **H** are:

MP: From A and $(A \Rightarrow B)$, infer B .

GEN: From A , infer $(\forall x . A)$, for any $x \in \mathcal{V}$.

More Metatheorems of FOL

- Deduction Theorem. $\Sigma \cup \{A\} \vdash_{\mathbf{H}} B$ implies $\Sigma \vdash_{\mathbf{H}} A \Rightarrow B$.
- Soundness Theorem. $\Sigma \vdash_{\mathbf{H}} A$ implies $\Sigma \models A$.
- Completeness Theorem. $\Sigma \models A$ implies $\Sigma \vdash_{\mathbf{H}} A$.
- Soundness and Completeness Theorem (second form).
 Σ is consistent in \mathbf{H} iff Σ is satisfiable.

Theories

- A **theory** in FOL is a pair $T = (L, \Gamma)$ where:
 1. L is a language of FOL.
 2. Γ is a set of sentences of L .
- **Examples:**
 - ▶ Theories of orders, lattices, and boolean algebras.
 - ▶ Theories of monoids and groups.
 - ▶ Presburger arithmetic.
 - ▶ First-order Peano arithmetic.
 - ▶ Theory of real closed fields.

The Theory of Boolean Algebras

- Let $\mathbf{BA} = (L, \Gamma)$ be the theory of FOL where L is defined below and Γ is the set of sentences of L on the next page.
- $L = (+, *, \neg, 0, 1, =)$ is a language of FOL such that $+$ and $*$ are binary function symbols, \neg is a unary function symbol, and 0 and 1 are individual constants.
- A **boolean algebra** is a model of \mathbf{BA} .
 - ▶ Named after the logician George Boole (1815-1864).
 - ▶ There are infinitely many nonisomorphic models of \mathbf{BA} .
 - ▶ If $(B, +, *, \neg, 0, 1)$ is a boolean algebra, then (B, \leq) is a complemented distributive lattice with a top and bottom where $a \leq b$ means $a = a * b \wedge a + b = b$.
- **Examples:**
 - ▶ $M_1 = (\{T, F\}, \vee, \wedge, \neg, F, T, \Leftrightarrow)$.
 - ▶ $M_2 = (\{S \mid S \subseteq U\}, \cup, \cap, \neg, \emptyset, U, =)$ where U is any set.
- \mathbf{BA} is used to model electronic circuits.

The Axioms of **BA**

Associativity Laws

$$\forall x, y, z . (x + y) + z = x + (y + z)$$

$$\forall x, y, z . (x * y) * z = x * (y * z)$$

Commutativity Laws

$$\forall x, y . x + y = y + x$$

$$\forall x, y . x * y = y * x$$

Distributive Laws

$$\forall x, y, z . x + (y * z) = (x + y) * (x + z)$$

$$\forall x, y, z . x * (y + z) = (x * y) + (x * z)$$

Identity Laws

$$\forall x . x + 0 = x$$

$$\forall x . x * 1 = x$$

Complement Laws

$$\forall x . x + \bar{x} = 1$$

$$\forall x . x * \bar{x} = 0$$

Theorems of BA

Idempotent Laws

$$\forall x . x + x = x$$

$$\forall x . x * x = x$$

Absorption Laws

$$\forall x, y . x + (x * y) = x$$

$$\forall x, y . x * (x + y) = x$$

De Morgan Laws

$$\forall x, y . \overline{x + y} = \bar{x} * \bar{y}$$

$$\forall x, y . \overline{x * y} = \bar{x} + \bar{y}$$

Laws of Zero and One

$$\forall x . x + 1 = 1$$

$$\forall x . x * 0 = 0$$

$$\overline{0} = 1$$

$$\overline{1} = 0$$

Law of Double Complement

$$\forall x . \overline{\bar{x}} = x$$

Peano Arithmetic

- **PA** = (L, Γ) is (second-order) Peano arithmetic (devised by G. Peano, 1889).
- L is a language of second-order logic with an individual constant symbol 0 and a unary function symbol S .
 - ▶ 0 is intended to represent the number **zero**.
 - ▶ S is intended to represent the **successor function**, i.e., $S(a)$ means $a + 1$.
- Γ is the following set of axioms:
 - ▶ **0 has no predecessor.** $\forall x . \neg(0 = S(x))$.
 - ▶ **S is injective.** $\forall x, y . S(x) = S(y) \Rightarrow x = y$.
 - ▶ **Induction principle.**
 $\forall P . (P(0) \wedge \forall x . P(x) \Rightarrow P(S(x))) \Rightarrow \forall x . P(x)$.
- $+$ and $*$ can be defined in **PA**.
- **PA** is **categorical**, i.e, it has exactly one model up to isomorphism (Dedekind, 1888).

First-Order Peano Arithmetic

- $\mathbf{PA}' = (L', \Gamma')$ is first-order Peano arithmetic.
- L' is a language of FOL with an individual constant symbol 0 , a unary function symbol S , and binary function symbols $+$ and $*$.
- Γ' is the following set of axioms:
 - ▶ $\forall x . \neg(S(x) = 0).$
 - ▶ $\forall x, y . S(x) = S(y) \Rightarrow x = y.$
 - ▶ $\forall x . x + 0 = x.$
 - ▶ $\forall x, y . x + S(y) = S(x + y).$
 - ▶ $\forall x . x * 0 = 0.$
 - ▶ $\forall x, y . x * S(y) = (x * y) + x.$
 - ▶ Each universal closure A of a formula of the form

$$(B[x \mapsto 0] \wedge (\forall x . B \Rightarrow B[x \mapsto S(x)])) \Rightarrow \forall x . B$$

where B is a formula of L' .

- \mathbf{PA}' is a noncategorical approximation of Peano arithmetic with infinitely many “nonstandard” models.