CAS 701 Fall 2004

06 First-Order Logic

Instructor: W. M. Farmer

Revised: 23 October 2004

What is First-Order Logic?

- **First-order logic** is the study of statements about individuals using functions, predicates, and quantification.
  - Extends propositional logic.
  - Quantification is allowed only over individuals—not over 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 $t_1, \ldots, t_n$ are terms of L, then $f(t_1, \ldots, t_n)$ 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 $t_1, \ldots, t_n$ are terms of L, then $p(t_1, \ldots, t_n)$ 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: Abbreviations

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

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, $I(p) : D^n \rightarrow \{\mathsf{t}, \mathsf{f}\}$ and $I(p)$ is total.
  - $I(=)$ is 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 $t = f(t_1, \ldots, t_n)$. Then $V_\varphi^M(t) = I(f)(V_\varphi^M(t_1), \ldots, V_\varphi^M(t_n))$.
4. Let $A = p(t_1, \ldots, t_n)$. Then $V_\varphi^M(A) = I(p)(V_\varphi^M(t_1), \ldots, V_\varphi^M(t_n))$.
5. Let $A = (\neg A')$. If $V_\varphi^M(A') = \mathsf{f}$, then $V_\varphi^M(A) = \mathsf{t}$; otherwise $V_\varphi^M(A) = \mathsf{f}$.
6. Let $A = (A_1 \Rightarrow A_2)$. If $V_\varphi^M(A_1) = \mathsf{t}$ and $V_\varphi^M(A_2) = \mathsf{f}$, then $V_\varphi^M(A) = \mathsf{f}$; otherwise $V_\varphi^M(A) = \mathsf{t}$.
7. Let $A = (\forall\, x \,.\, A')$. If $V_{\varphi[x \mapsto d]}^M(A') = \mathsf{t}$ for all $d \in D$, then $V_\varphi^M(A) = \mathsf{t}$; otherwise $V_\varphi^M(A) = \mathsf{f}$.

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)

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)

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

A10: $\forall\, x_1, \ldots, x_n, y_1, \ldots, y_n \,.\, (x_1 = y_1 \wedge \cdots \wedge x_n = y_n)$
$\Rightarrow (p(x_1, \ldots, x_n) \Leftrightarrow p(y_1, \ldots, y_n))$
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 **H** iff Σ is satisfiable.

Semantics vs. Syntax

Semantics	**Syntax**
A is valid $\models A$	A is a theorem in **P** $\vdash_{\mathbf{P}} A$
A is valid in T $T \models A$	A is a theorem of T in **P** $T \vdash_{\mathbf{P}} A$
T is satisfiable	T is consistent in **P**

- By Gödel's Completeness Theorem, the semantic and syntactic notions for first-order logic are equivalent

- The problem whether or not $T \models A$ is true can be solved by either:

  1. **Proof**: Showing $T \vdash_{\mathbf{P}} A$ for some sound proof system **P** or
  2. **Counterexample**: Showing $M \models \neg A$ for some model M of T.