07—Predicate Logic II

UIT2206: The Importance of Being Formal

Martin Henz

March 5, 2014

Generated on Wednesday 5th March, 2014, 10:05

Terms

$$t ::= \ x \mid c \mid f(t, \ldots, t)$$

where

- x ranges over a given set of variables $\mathcal{V}$,
- c ranges over nullary function symbols in $\mathcal{F}$, and
- f ranges over function symbols in $\mathcal{F}$ with arity $n > 0$.

Formulas

$$\begin{aligned}\phi \quad ::= \quad & P(t, \ldots, t) \mid (\neg\phi) \mid (\phi \land \phi) \mid (\phi \lor \phi) \mid \\ & (\phi \rightarrow \phi) \mid (\forall x \phi) \mid (\exists x \phi)\end{aligned}$$

where

- $P \in \mathcal{P}$ is a predicate symbol of arity $n \geq 0$,
- t are terms over $\mathcal{F}$ and $\mathcal{V}$, and
- x are variables in $\mathcal{V}$.

Models

Definition

Let $\mathcal{F}$ contain function symbols and $\mathcal{P}$ contain predicate symbols. A model $\mathcal{M}$ for $(\mathcal{F}, \mathcal{P})$ consists of:

1. A non-empty set A, the *universe*;
2. for each nullary function symbol $f \in \mathcal{F}$ a concrete element $f^{\mathcal{M}} \in A$;
3. for each $f \in F$ with arity $n > 0$, a concrete function $f^{\mathcal{M}} : A^n \to A$;
4. for each $P \in \mathcal{P}$ with arity $n > 0$, a function $P^{\mathcal{M}} : U^n \to \{F, T\}$.
5. for each $P \in \mathcal{P}$ with arity $n = 0$, a value from $\{F, T\}$.

Example

Let $\mathcal{F} = \{e, \cdot\}$ and $\mathcal{P} = \{\leq\}$.
Let model $\mathcal{M}$ for $(\mathcal{F}, \mathcal{P})$ be defined as follows:

1. Let A be the set of binary strings over the alphabet $\{0, 1\}$;
2. let $e^{\mathcal{M}} = \epsilon$, the empty string;
3. let $\cdot^{\mathcal{M}}$ be defined such that $s_1 \cdot^{\mathcal{M}} s_2$ is the concatenation of the strings s_1 and s_2; and
4. let $\leq^{\mathcal{M}}$ be defined such that $s_1 \leq^{\mathcal{M}} s_2$ iff s_1 is a prefix of s_2.

Example (continued)

1. Let A be the set of binary strings over the alphabet $\{0, 1\}$;
2. let $e^{\mathcal{M}} = \epsilon$, the empty string;
3. let $\cdot^{\mathcal{M}}$ be defined such that $s_1 \cdot^{\mathcal{M}} s_2$ is the concatenation of the strings s_1 and s_2; and
4. let $\leq^{\mathcal{M}}$ be defined such that $s_1 \leq^{\mathcal{M}} s_2$ iff s_1 is a prefix of s_2.

Some Elements of A

- 10001
- ϵ
- $1010 \cdot^{\mathcal{M}} 1100 = 10101100$
- $000 \cdot^{\mathcal{M}} \epsilon = 000$

Equality Revisited

Interpretation of equality

Usually, we require that the equality predicate $=$ is interpreted as same-ness.

Extensionality restriction

This means that allowable models are restricted to those in which $a =^{\mathcal{M}} b$ holds if and only if a and b are the same elements of the model's universe.

Example (continued)

1. Let A be the set of binary strings over the alphabet $\{0, 1\}$;
2. let $e^{\mathcal{M}} = \epsilon$, the empty string;
3. let $\cdot^{\mathcal{M}}$ be defined such that $s_1 \cdot^{\mathcal{M}} s_2$ is the concatenation of the strings s_1 and s_2; and
4. let $\leq^{\mathcal{M}}$ be defined such that $s_1 \leq^{\mathcal{M}} s_2$ iff s_1 is a prefix of s_2.

Equality in $\mathcal{M}$

- $000 =^{\mathcal{M}} 000$
- $001 \neq^{\mathcal{M}} 100$

How To Handle Free Variables?

Idea

We can give meaning to formulas with free variables by providing an environment (lookup table) that assigns variables to elements of our universe:

$$I : \mathcal{V} \to A.$$

Environment extension

We define environment extension such that $I[x \mapsto a]$ is the environment that maps x to a and any other variable y to $I(y)$.

Satisfaction Relation

The model $\mathcal{M}$ satisfies ϕ with respect to environment l, written $\mathcal{M} \models_l \phi$:

- in case ϕ is of the form $P(t_1, t_2, \ldots, t_n)$, if $a_1, a_2, \ldots, a_n$ are the results of evaluating $t_1, t_2, \ldots, t_n$ with respect to l, and if $P^{\mathcal{M}}(a_1, a_2, \ldots, a_n) = T$;
- in case ϕ is of the form P, if $P^{\mathcal{M}} = T$;
- in case ϕ has the form $\forall x \psi$, if the $\mathcal{M} \models_{l[x \mapsto a]} \psi$ holds for all $a \in A$;
- in case ϕ has the form $\exists x \psi$, if the $\mathcal{M} \models_{l[x \mapsto a]} \psi$ holds for some $a \in A$;

Satisfaction Relation (continued)

- in case ϕ has the form $\neg\psi$, if $\mathcal{M} \models_l \psi$ does not hold;
- in case ϕ has the form $\psi_1 \vee \psi_2$, if $\mathcal{M} \models_l \psi_1$ holds or $\mathcal{M} \models_l \psi_2$ holds;
- in case ϕ has the form $\psi_1 \wedge \psi_2$, if $\mathcal{M} \models_l \psi_1$ holds and $\mathcal{M} \models_l \psi_2$ holds; and
- in case ϕ has the form $\psi_1 \rightarrow \psi_2$, if $\mathcal{M} \models_l \psi_2$ holds whenever $\mathcal{M} \models_l \psi_1$ holds.

Satisfaction of Closed Formulas

If a formula ϕ has no free variables, we call ϕ a *sentence*.
$\mathcal{M} \models_I \phi$ holds or does not hold regardless of the choice of I.
Thus we write $\mathcal{M} \models \phi$ or $\mathcal{M} \not\models \phi$.

Semantic Entailment and Satisfiability

Let Γ be a possibly infinite set of formulas in predicate logic and ψ a formula.

Entailment

$\Gamma \models \psi$ iff for all models $\mathcal{M}$ and environments l, whenever $\mathcal{M} \models_l \phi$ holds for all $\phi \in \Gamma$, then $\mathcal{M} \models_l \psi$.

Satisfiability of Formulas

ψ is satisfiable iff there is some model $\mathcal{M}$ and some environment l such that $\mathcal{M} \models_l \psi$ holds.

Satisfiability of Formula Sets

Γ is satisfiable iff there is some model $\mathcal{M}$ and some environment l such that $\mathcal{M} \models_l \phi$, for all $\phi \in \Gamma$.

Semantic Entailment and Satisfiability

Let Γ be a possibly infinite set of formulas in predicate logic and ψ a formula.

Validity

ψ is valid iff for all models $\mathcal{M}$ and environments l, we have $\mathcal{M} \models_l \psi$.

The Problem with Predicate Logic

Entailment ranges over models

Semantic entailment between sentences: $\phi_1, \phi_2, \ldots, \phi_n \models \psi$ requires that in *all* models that satisfy $\phi_1, \phi_2, \ldots, \phi_n$, the sentence ψ is satisfied.

How to effectively argue about all possible models?

Usually the number of models is infinite; it is very hard to argue on the semantic level in predicate logic.

Idea from propositional logic

Can we use natural deduction for showing entailment?

Natural Deduction for Predicate Logic

Relationship between propositional and predicate logic

If we consider propositions as nullary predicates, propositional logic is a sub-language of predicate logic.

Inheriting natural deduction

We can translate the rules for natural deduction in propositional logic directly to predicate logic.

Example

$$\frac{\phi \qquad \psi}{\phi \wedge \psi}[\wedge i]$$

Built-in Rules for Equality

$$\frac{}{t = t}[= i] \qquad \frac{t_1 = t_2 \qquad [x \Rightarrow t_1]\phi}{[x \Rightarrow t_2]\phi}[= e]$$

Properties of Equality

We show:

$$f(x) = g(x) \vdash h(g(x)) = h(f(x))$$

using

$$\frac{}{t = t}[= i] \qquad \frac{t_1 = t_2 \qquad [x \Rightarrow t_1]\phi}{[x \Rightarrow t_2]\phi}[= e]$$

1	$f(x) = g(x)$	premise
2	$h(f(x)) = h(f(x))$	$= i$
3	$h(g(x)) = h(f(x))$	$= e$ 1,2

Elimination of Universal Quantification

$$\frac{\forall x \phi}{[x \Rightarrow t]\phi}[\forall x\ e]$$

Once you have proven $\forall x \phi$, you can replace x by any term t in ϕ, provided that t is free for x in ϕ.

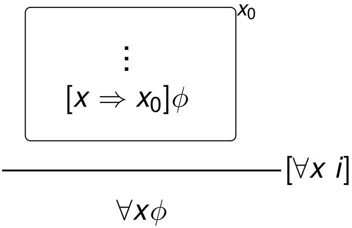
Example

$$\frac{\forall x \phi}{[x \Rightarrow t]\phi}[\forall x\ e]$$

We prove: $S(g(john)), \forall x(S(x) \rightarrow \neg L(x)) \vdash \neg L(g(john))$

1	$S(g(john))$	premise
2	$\forall x(S(x) \rightarrow \neg L(x))$	premise
3	$S(g(john)) \rightarrow \neg L(g(john))$	$\forall x\ e$ 2
4	$\neg L(g(john))$	$\rightarrow e$ 3,1

Introduction of Universal Quantification



$$\frac{\boxed{\begin{array}{c}\vdots\\ [x \Rightarrow x_0]\phi\end{array}}^{\,x_0}}{\forall x \phi}\,[\forall x\ i]$$

If we manage to establish a formula ϕ about a fresh variable x_0, we can assume $\forall x \phi$.

The variable x_0 must be *fresh*; we cannot introduce the same variable twice in nested boxes.

Example

$$\forall x(P(x) \to Q(x)), \forall x P(x) \vdash \forall x Q(x) \text{ via } \frac{\boxed{\begin{array}{c}\vdots\\ [x \Rightarrow x_0]\phi\end{array}}^{\,x_0}}{\forall x \phi}$$

1	$\forall x(P(x) \to Q(x))$	premise	
2	$\forall x P(x)$	premise	
3	$P(x_0) \to Q(x_0)$	$\forall x\ e$ 1	x_0
4	$P(x_0)$	$\forall x\ e$ 2	
5	$Q(x_0)$	$\to e$ 3,4	
6	$\forall x Q(x)$	$\forall x\ i$ 3–5	

Introduction of Existential Quantification

$$\frac{[x \Rightarrow t]\phi}{\exists x \phi}\,[\exists x\ i]$$

In order to prove $\exists x \phi$, it suffices to find a term t as "witness", provided that t is free for x in ϕ.

Example

$$\forall x \phi \vdash \exists x \phi$$

Recall: Definition of Models

A model $\mathcal{M}$ for $(\mathcal{F}, \mathcal{P})$ consists of:

1. A *non-empty* set U, the *universe*;
2. ...

Remark

Compare this with Traditional Logic.

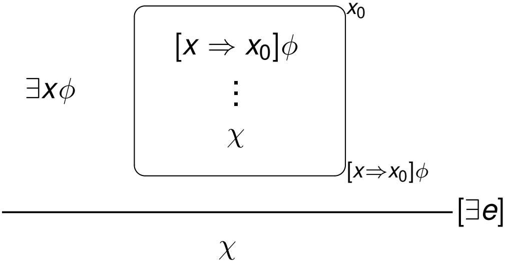
Because U must not be empty, we should be able to prove the sequent above.

Example (continued)

$$\forall x \phi \vdash \exists x \phi$$

1	$\forall x \phi$	premise
2	$[x \Rightarrow x]\phi$	$\forall x$ *e* 1
3	$\exists x \phi$	$\exists x$ *i* 2

Elimination of Existential Quantification



x_0

$\exists x \phi$

$[x \Rightarrow x_0]\phi$

$\vdots$

χ

$[x \Rightarrow x_0]\phi$

$[\exists e]$

χ

Making use of $\exists$

If we know $\exists x \phi$, we know that there exist at least one object x for which ϕ holds. We call that element x_0, and assume $[x \Rightarrow x_0]\phi$. Without assumptions on x_0, we prove χ (x_0 not in χ).

Example

$$\forall x(P(x) \rightarrow Q(x)), \exists x P(x) \vdash \exists x Q(x)$$

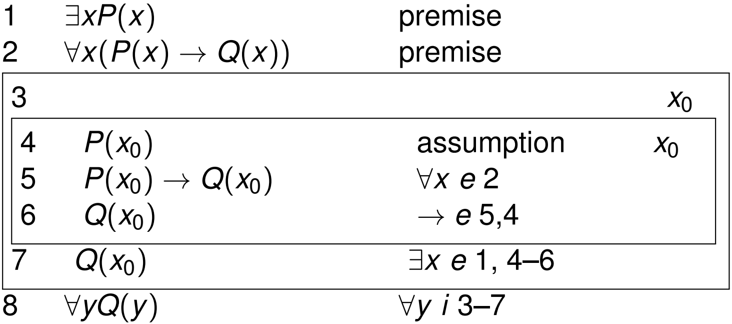
1	$\forall x(P(x) \rightarrow Q(x))$	premise	
2	$\exists x P(x)$	premise	
3	$P(x_0)$	assumption	x_0
4	$P(x_0) \rightarrow Q(x_0)$	$\forall x$ *e* 1	
5	$Q(x_0)$	$\rightarrow$ *e* 4,3	
6	$\exists x Q(x)$	$\exists x$ *i* 5	
7	$\exists x Q(x)$	$\exists x$ *e* 2,3–6	

Note that $\exists x Q(x)$ within the box does not contain x_0, and therefore can be “exported” from the box.

Another Example

1	$\forall x(Q(x) \rightarrow R(x))$	premise	
2	$\exists x(P(x) \wedge Q(x))$	premise	
3	$P(x_0) \wedge Q(x_0)$	assumption	x_0
4	$Q(x_0) \rightarrow R(x_0)$	$\forall x\ e$ 1	
5	$Q(x_0)$	$\wedge e_2$ 3	
6	$R(x_0)$	$\rightarrow e$ 4,5	
7	$P(x_0)$	$\wedge e_1$ 3	
8	$P(x_0) \wedge R(x_0)$	$\wedge i$ 7, 6	
9	$\exists x(P(x) \wedge R(x)$	$\exists x\ i$ 8	
10	$\exists x(P(x) \wedge R(x))$	$\exists x\ e$ 2,3–9	

Variables must be fresh! This is not a proof!



1	$\exists x P(x)$	premise	
2	$\forall x (P(x) \to Q(x))$	premise	
3			x_0
4	$P(x_0)$	assumption	x_0
5	$P(x_0) \to Q(x_0)$	$\forall x\ e$ 2	
6	$Q(x_0)$	$\to e$ 5,4	
7	$Q(x_0)$	$\exists x\ e$ 1, 4–6	
8	$\forall y Q(y)$	$\forall y\ i$ 3–7	

1. Review: Syntax and Semantics of Predicate Logic
2. Semantics: Satisfaction, Entailment, Satisfiability
3. Proof Theory
4. Equivalences
    - Quantifier Equivalences
    - Soundness and Completeness
    - Undecidability, Compactness

Equivalences

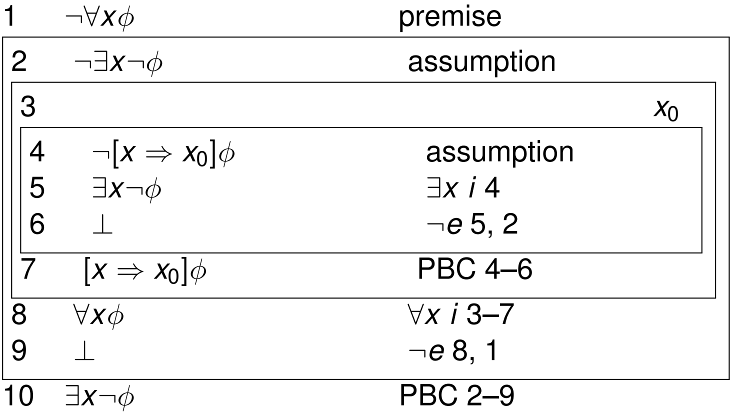
Two-way-provable

We write $\phi \dashv\vdash \psi$ iff $\phi \vdash \psi$ and also $\psi \vdash \phi$.

Some simple equivalences

$$\begin{aligned}
\neg\forall x\phi &\dashv\vdash \exists x\neg\phi \\
\neg\exists x\phi &\dashv\vdash \forall x\neg\phi \\
\forall x\forall y\phi &\dashv\vdash \forall y\forall x\phi \\
\exists x\exists y\phi &\dashv\vdash \exists y\exists x\phi \\
\forall x\phi \wedge \forall x\psi &\dashv\vdash \forall x(\phi \wedge \psi) \\
\exists x\phi \vee \exists x\psi &\dashv\vdash \exists x(\phi \vee \psi)
\end{aligned}$$

$\neg\forall x\phi \vdash \exists x\neg\phi$



1	$\neg\forall x\phi$	premise
2	$\neg\exists x\neg\phi$	assumption
3		x_0
4	$\neg[x \Rightarrow x_0]\phi$	assumption
5	$\exists x\neg\phi$	$\exists x\ i$ 4
6	$\bot$	$\neg e$ 5, 2
7	$[x \Rightarrow x_0]\phi$	PBC 4–6
8	$\forall x\phi$	$\forall x\ i$ 3–7
9	$\bot$	$\neg e$ 8, 1
10	$\exists x\neg\phi$	PBC 2–9

$\exists x \exists y \phi \vdash \exists y \exists x \phi$

Assume that x and y are different variables.

1	$\exists x \exists y \phi$	premise	
2	$[x \Rightarrow x_0](\exists y \phi)$	assumption	x_0
3	$\exists y([x \Rightarrow x_0]\phi$	def of subst (x, y different)	
4	$[y \Rightarrow y_0][x \Rightarrow x_0]\phi$	assumption	y_0
5	$[x \Rightarrow x_0][y \Rightarrow y_0]\phi$	def of subst (x, y, x_0, y_0 different)	
6	$\exists x[y \rightarrow y_0]\phi$	$\exists x$ *i* 5	
7	$\exists y \exists x \phi$	$\exists y$ *i* 6	
8	$\exists y \exists x \phi$	$\exists y$ *e* 3, 4–7	
9	$\exists y \exists x \phi$	$\exists x$ *e* 1, 2–8	

More Equivalences

Assume that *x* is not free in ψ

$$
\begin{aligned}
\forall x \phi \wedge \psi &\quad \dashv\vdash \quad \forall x(\phi \wedge \psi) \\
\forall x \phi \vee \psi &\quad \dashv\vdash \quad \forall x(\phi \vee \psi) \\
\exists x \phi \wedge \psi &\quad \dashv\vdash \quad \exists x(\phi \wedge \psi) \\
\exists x \phi \vee \psi &\quad \dashv\vdash \quad \exists x(\phi \vee \psi)
\end{aligned}
$$

Central Result of Natural Deduction

$$\phi_1, \ldots, \phi_n \models \psi$$

iff

$$\phi_1, \ldots, \phi_n \vdash \psi$$

proven by Kurt Gödel, in 1929 in his doctoral dissertation

Recall: Decidability

Decision problems

A *decision problem* is a question in some formal system with a yes-or-no answer.

Decidability

Decision problems for which there is an algorithm that returns "yes" whenever the answer to the problem is "yes", and that returns "no" whenever the answer to the problem is "no", are called *decidable*.

Decidability of satisfiability

The question, whether a given propositional formula is satisifiable, is decidable.

Undecidability of Predicate Logic

Theorem

The decision problem of validity in predicate logic is undecidable: no program exists which, given any language in predicate logic and any formula ϕ in that language, decides whether $\models \phi$.

Proof sketch

- Establish that the Post Correspondence Problem (PCP) is undecidable
- Translate an arbitrary PCP, say C, to a formula ϕ.
- Establish that $\models \phi$ holds if and only if C has a solution.
- Conclude that validity of predicate logic formulas is undecidable.

Next Week

- midterm
- Modal Logic I