Mathematical Foundations of computer science

Lecture 7: First-order predicate logic (continued)

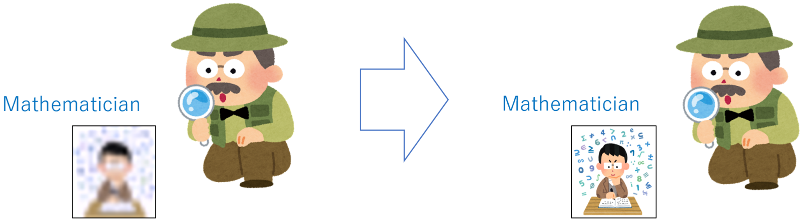
Speaker: Toru Takisaka

April 16, 2024

Recap

Current goal: formalize Mathematician with higher resolution

- We construct atomic propositions from terms / predicates
- Today: *what are truth values of predicate formulas?*



Recap: from propositional logic to predicate logic

In Lecture 2, we said:

Formula = statement whose "correctness" can be argued

- "Dr. Takisaka is a professor"
- "Roses are blue"
- "$1 + 1 = 3$"

These formulas are *atomic*, i.e., they cannot be split into multiple formulas.

...but they are often too crude as the minimal parts of formulas.

In fact, they are made with subjects and predicates:

Recap: terms and predicates

Terms = objects whose properties are argued

- "n (which can be a number)", " φ (which can be a propositional formula)", ...
  $\rightarrow$ Variables... represent undetermined objects
- "Dr. Takisaka", "Roses", "0"...
  $\rightarrow$ Constant symbols... designate specific objects
- "$1 + 1$", "$x + y$", "The father of Dr. Takisaka",...
  $\rightarrow$ Terms made via function symbols

Predicates are just symbols with their arities (num. of inputs).

- "is a professor" is a unary predicate
- "is blue" is a unary predicate
- "$=$" is a binary predicate

Recap: predicate formulas

Now "$t_1, \ldots, t_k$ satisfy P" is written as $P(t_1, \ldots, t_k)$, where $t_1, \ldots, t_k$ are terms and P is a k-ary predicate.

Definition (first-order predicate formula over language L)

Base case (defines atomic, or equivalently base, formulas):

- Given terms t_1, ..., t_k and a predicate $P \in L$ of relation k, the expression $P(t_1, \ldots, t_k)$ is a formula.
- The expression $(t_1 = t_2)$ is a formula, t_1 and t_2 are terms.

Inductive step:

- If ϕ and ψ are formulas, then so are $(\phi \,\&\, \psi)$, $\neg\phi$, and $(\phi \;\vee\; \psi)$.
- If ϕ is a formula and x is a variable, then $\forall x \phi$ and $\exists x \phi$ are formulas.

As in the propositional logic case, $(\phi \to \psi)$ stands for $(\neg\phi \vee \psi)$.

Examples

- "Dr. Takisaka is a professor" could be formalized as a formula

$$IsProf(\text{Dr. Takisaka}),$$

  where, *IsProf* is a unary predicate symbol and "Dr. Takisaka" is a constant symbol.

- "Any professor in China is over 20years old" could be formalized as a formula

$$\forall x.\Big((IsProf(x) \land IsInChina(x)) \to Over20(x)\Big),$$

  where, *IsProf*, *IsInChina*, and *Over*20 are unary predicate symbols.

Examples

- "for any n, either n or $n+1$ is an odd number" could be formalized as a formula

$$\forall n.\left(\mathit{IsOdd}(n) \vee \mathit{IsOdd}(S(n)) \right),$$

where, S is a unary function symbol, and *IsOdd* is a unary predicate symbol.

  - S...*successor* function symbol, which represents "the next number"

- The *twin prime conjecture* could be formalized as a formula

$$\exists x.\forall y.\left(\mathit{prime}(y) \wedge \mathit{prime}(y+2) \rightarrow y \leq x \right),$$

where, *prime* is a unary predicate symbol, and $\leq$ is a binary predicate symbol.

Recap: Algebraic structures

Definition

An **algebraic structure (or simply structure)** $\mathcal{A}$ is a tuple:

$$(A; P_0^{m_0}, \ldots, P_k^{m_k}, f_0^{n_0}, \ldots, f_t^{n_t}),$$

where:

- A is a non-empty set called the **domain** of the structure,
- Each $P_i^{m_i}$, $i = 1, \ldots, k$, is a relation of arity m_i on A, and
- Each $f_j^{n_j}$, $j = 1, \ldots, t$, is an operation of arity n_j on A.

- Domain = the set of objects we want to talk about
- relations = the meaning of predicate symbols
- functions = the meaning of function symbols

Examples

"Any professor in China is over 20years old" could be formalized as a formula

$$\forall x. \Big((IsProf(x) \wedge IsInChina(x)) \to Over20(x) \Big).$$

The language $L = (IsProf, IsInChina, Over20)$ can write it.

The structure $\mathcal{A} = (H; IsProf_{\mathcal{A}}, IsInChina_{\mathcal{A}}, Over20_{\mathcal{A}})$ would be an intended structure, where

- H is the set of humans (say, who were alive at a certain timestamp),
- $IsProf_{\mathcal{A}} \subseteq H$ is the set of professors in H (say, at that timestamp),
- $IsInChina_{\mathcal{A}} \subseteq H$ is the set of residents in China in H, and
- $Over20_{\mathcal{A}} \subseteq H$ is the set of humans in H that are over 20 years old.

Examples

"for any n, either n or $n+1$ is an odd number" could be formalized as a formula

$$\forall n. \Big(IsOdd(n) \vee IsOdd(S(n)) \Big).$$

The language $L = (S, IsOdd)$ can write it.

An intended structure would be $\mathcal{A} = (\mathbb{N}; S_{\mathcal{A}}, IsOdd_{\mathcal{A}})$, where

- $\mathbb{N}$ is the set of natural numbers,
- $S_{\mathcal{A}} : \mathbb{N} \to \mathbb{N}$ is a function such that $S_{\mathcal{A}}(n) = n + 1$, and
- $IsOdd_{\mathcal{A}} \subseteq \mathbb{N}$ is a unary relation such that

$$IsOdd_{\mathcal{A}} = \{ n \in \mathbb{N} \mid n \neq m + m \quad \text{for any } m \in \mathbb{N} \}$$

Examples

The *twin prime conjecture* could be formalized as a formula

$$\exists x. \forall y. \Big(prime(y) \wedge prime(S(S(y))) \rightarrow y \leq x \Big).$$

The language $L = (S, \leq, prime)$ can write it.

We can consider a structure $\mathcal{A} = (\mathbb{N}; S_{\mathcal{A}}, \leq_{\mathcal{A}}, prime_{\mathcal{A}})$, where

- $\mathbb{N}$ and $S_{\mathcal{A}}$ are as in the previous slide,
- $\leq_{\mathcal{A}} \subseteq \mathbb{N}^2$ is a binary relation such that

$$\leq_{\mathcal{A}} = \{(n, m) \in \mathbb{N}^2 \mid n + n' = m \quad \text{for some } n' \in \mathbb{N}\},$$

- $prime_{\mathcal{A}} \subseteq \mathbb{N}$ is a unary relation such that

$$prime_{\mathcal{A}} = \{n \in \mathbb{N} \mid \text{for any } m_1, m_2 \in \mathbb{N}, \\ \text{if } m = m_1 \cdot m_2 \text{ then } m_1 = 1 \text{ or } m_2 = 1\}.$$

Now we are almost ready to formalize what we mean by saying "a statement φ is true" in predicate logic.

We still need a couple of additional setup to this end;
we will explain these things while formalizing above.

free variables and bound variables

After specifying a structure, the truth of some formulas are still undetermined.

- Consider a formula $\mathit{IsOdd}(n)$ with the structure as given in p10. Thus it could read "a natural number *n* is odd".

  $\rightarrow$ We cannot determine the truth of this formula, as it depends on what *n* is.

- Consider a formula $\exists n.\mathit{IsOdd}(n)$ with the same structure. Thus it could read "there exists an odd natural number *n*".

  $\rightarrow$ the truth of this formula should be determined, because the meaning of *n* is *bound* by the prefix "there exists".

n in the first formula is **free**; *n* in the second is **bound**.

free variables and bound variables

A **subformula** of a formula φ is a formula that appears in φ (see textbook for a formal definition).

Ex.) Subformulas of $\psi \equiv (\exists x.P(x) \vee \neg Q(y))$ are

$$\psi,\ \exists x.P(x),\ P(x),\ \neg Q(y),\ \text{and}\ \ Q(y).$$

An occurrence of a variable x in φ is **bound in** φ if there is a subformula φ' of φ as below; otherwise, it is **free in** φ.

- φ' is either of the form $\varphi' \equiv \forall x.\psi$ or $\varphi' \equiv \exists x.\psi$, and
- the x under consideration occurs in ψ.

Ex.) In ψ above, (the unique occurrence of) the variable x in ψ is bound, and y is free.

A formula is called a **sentence** if it has no free variable.

Truth of atomic sentences

An atomic formula $P(t_1, \ldots, t_k)$ is a sentence if and only if there is no occurrence of a variable in $t_1, \ldots, t_k$.

Terms that do not contain variables are called **ground terms**; once a structure $\mathcal{A}$ is fixed, a ground term t designates an element $t^{\mathcal{A}}$ in the domain of $\mathcal{A}$ (see textbook for a formal definition).

Also recall $\mathcal{A}$ gives the meaning of P by the relation $P^{\mathcal{A}}$ (Intuitively, $\vec{t} \in P^{\mathcal{A}}$ means "$\vec{t}$ satisfies P under $\mathcal{A}$").

We say $P(t_1, \ldots, t_k)$ is true under a structure $\mathcal{A}$ if and only if

$$(t_1^{\mathcal{A}}, \ldots, t_k^{\mathcal{A}}) \in P^{\mathcal{A}}.$$

Examples

For a constant symbol "Dr. Takisaka" and a unary function symbol *FatherOf*, the following is a ground term:

$$\mathit{FatherOf}(\text{Dr. Takisaka})$$

Once a structure is fixed, it designates an element of domain (under an appropriate structure, it would designate the father of Dr. Takisaka).

For a unary predicate *IsProf*, the following is a sentence:

$$\mathit{IsProf}(\mathit{FatherOf}(\text{Dr. Takisaka}))$$

If the structure of *IsProf* is as in P9, then the sentence above is true if and only if the father of Dr. Takisaka is a professor.

Examples

Consider a language $(\overline{0}, S, +, \leq)$, which typically represents the arithmetic over natural numbers:

- $\overline{0}$ is a constant symbol
- S is a unary function symbol
- $+$ is a binary function symbol (write $n + m$ instead of $+(n, m)$)
- $\leq$ is a binary predicate symbol (write $n \leq m$ instead of $\leq (n, m)$)

$\overline{0}, S(\overline{0}), S(S(\overline{0})), S(S(\overline{0})) + S(S(\overline{0}))$ are all ground terms
(under the standard structure, they designate natural numbers $0, 1, 2, 4$, respectively).

$\overline{0} \leq S(\overline{0}),\ S(\overline{0}) + S(\overline{0}) = S(S(\overline{0}))$ are atomic sentences
(both are true under the standard structure).

Truth of $t_1 = t_2$, and composite formulas

For ground terms t_1 and t_2, the formula $t_1 = t_2$ is true under $\mathcal{A}$ if and only if $t_1^{\mathcal{A}}$ and $t_2^{\mathcal{A}}$ are the same element of the domain.

- Or alternatively, we can say "$=$" is a binary predicate that must always be interpreted as the equality relation:

$$=^{\mathcal{A}} \;=\; \{(x, x) \mid x \in A\} \qquad (A \text{ is the domain of } \mathcal{A})$$

The truth value of $\neg\varphi$, $(\varphi \vee \psi)$ and $(\varphi \wedge \psi)$ are defined in the similar way to propositional logic.

Substitution of variables in a formula

When we are interested in a formula φ and variables $x_1, \ldots, x_k$ that freely occur in φ, we also write $\varphi(x_1, \ldots, x_k)$ to denote φ.

We write $\varphi(t)$ to denote the formula obtained by substituting each **free** occurrence of x in $\varphi(x)$ with the term t.

- If $\varphi(x) \equiv \forall x.P(x) \vee Q(x)$, then $\varphi(t) \equiv \forall x.P(x) \vee Q(t)$, because x in $\forall x.P(x)$ is bound.

CAUTION: The term t may have free variables in it. These variables must not be bound as a result of substitution.

- If $\psi(x) \equiv \forall y.(P(x) \vee Q(y))$ and $t \equiv y$, then you cannot do $\psi(t)$ (you can take $\psi' \equiv \forall z.(P(x) \vee Q(z))$ instead and do $\psi'(t)$).

Truth of $\forall x.\varphi(x)$ and $\exists x.\varphi(x)$

Intuitively, we would like to say $\forall x.\varphi(x)$ is true under a structure $\mathcal{A}$ if and only if "$\varphi(a)$ is true" for all element a of the domain. But what is $\varphi(a)$? a is not a term, so $\varphi(a)$ is not a formula.

- Consider the language $(\overline{0}, S, +, \leq)$. The formula $\forall n.n \leq n$ should be clearly true under the standard structure $\mathcal{A}$, because we have $0 \leq^{\mathcal{A}} 0$, $1 \leq^{\mathcal{A}} 1$, $2 \leq^{\mathcal{A}} 2$, and so on.

  But our language cannot write formulas like "$1 \leq 1$" and "$2 \leq 2$" because 1 and 2 are not in it.

Solution: Augment the language with new constant symbols.

Truth of $\forall x.\varphi(x)$ and $\exists x.\varphi(x)$

If our language is $L = (\overline{0}, S, +, \leq)$ and we consider the standard structure $\mathcal{A}$ (whose domain is $\mathbb{N}$), then

- consider $L_{\mathcal{A}} = (\overline{0}, S, +, \leq, 0^*, 1^*, 2^*, \ldots)$;
- consider the structure $\mathcal{A}^*$ for $L_{\mathcal{A}}$ that assigns 0 to 0^*, 1 to 1^*..., and assigns the same functions as $\mathcal{A}$ to $\overline{0}, S, +, \leq$;
- say $\forall n.n \leq n$ is true under $\mathcal{A}$ if and only if $n^* \leq n^*$ is true under $\mathcal{A}^*$ for any $n \in \mathbb{N}$ (indeed it is, check it).

Truth of $\forall x.\varphi(x)$ and $\exists x.\varphi(x)$

More generally, let $\forall x.\varphi(x)$ and $\exists x.\varphi(x)$ be sentences over a language L, let $\mathcal{A}$ be its structure, and let A be the domain of $\mathcal{A}$. The truth of these sentences are defined in the following way:

- Define a new language $L_{\mathcal{A}}$ by adding a new constant symbol a^* to L, for each $a \in A$.
- Let $\mathcal{A}^*$ be a structure for $L_{\mathcal{A}}$ that assigns a to a^*, for each $a \in A$; and assigns the same function/relation as $\mathcal{A}$ for each function/predicate symbol in L.
- say $\forall x.\varphi(x)$ is true under $\mathcal{A}$ if and only if $\varphi(a^*)$ is true under $\mathcal{A}^*$ for any $a \in A$.
  say $\exists x.\varphi(x)$ is true under $\mathcal{A}$ if and only if $\varphi(a^*)$ is true under $\mathcal{A}^*$ for some $a \in A$.