

# Predicate Calculus

(First-Order Logic)

## Syntax

A *first-order vocabulary* (or just *vocabulary* or *language*)  $\mathcal{L}$  is specified by the following:

- 1) For each  $n \in \mathbb{N}$  a set of  $n$ -ary function symbols (possibly empty). We use  $f, g, h, \dots$  and also  $+$ ,  $\cdot$ ,  $s$  as metasymbols for function symbols. A zero-ary function symbol is called a constant symbol.
- 2) For each  $n \geq 0$ , a set of  $n$ -ary predicate symbols (must be non-empty for some  $n$ ). We use  $P, Q, R, \dots$  and also  $<$ ,  $\leq$ ,  $=$  as metasymbols for predicate symbols. A zero-ary predicate symbol is the same as a propositional atom.

In addition, the following symbols are available to build first-order formulas:

- 1) An infinite set of variables. We use  $x, y, z, \dots$  and sometimes  $a, b, c, \dots$  as metasymbols for variables. (Generally distinct letters  $x, y, z$  stand for distinct variables.)
- 2) connectives  $\neg, \wedge, \vee$  (not, and, or)
- 3) quantifiers  $\forall, \exists$  (for all, there exists)
- 4)  $(, )$  (parentheses)

Terms and Formulas are built from these together with the function and predicate symbols from  $\mathcal{L}$ , as described below.

The standard vocabulary of arithmetic is

$$\mathcal{L}_A = [0, s, +, \cdot ; =]$$

- 0 constant (zero-ary function symbol)
- $s$  unary function symbol
- $+, \cdot$  binary function symbols
- $=$  binary predicate symbol

*Terms* (or *expressions*) are certain strings built from variables and function symbols, and are intended to represent objects in the universe of discourse.

**Definition of an  $\mathcal{L}$ -term** (Here  $\mathcal{L}$  is a first-order vocabulary):

- 1) Every variable is a term.
- 2) If  $f$  is an  $n$ -ary function symbol of  $\mathcal{L}$  and  $t_1, \dots, t_n$  are  $\mathcal{L}$ -terms then  $ft_1 \dots t_n$  is an  $\mathcal{L}$ -term.

We will drop mention of  $\mathcal{L}$  when it is not important, or clear from context.

Recall that a 0-ary function symbol is called a constant symbol (or sometimes just a *constant*). We use  $e$  as a metasymbol for constants. Also 0 and 1 are constants. Note that all constants in  $\mathcal{L}$  are  $\mathcal{L}$ -terms.

**Examples of  $\mathcal{L}$ -terms** (where  $f$  is binary and  $g$  is unary):  
 $fgex, fxy, gfege$ . These are parsed  $f(g(e), x), f(x, y), g(f(e, g(e)))$  respectively.

**Unique Readability Theorem for Terms:** If terms  $ft_1 \dots t_k$  and  $fu_1 \dots u_\ell$  are syntactically equal, then  $k = \ell$  and  $t_i =_{syn} u_i, 1 \leq i \leq k$ .

**Proof:** Similar to the Unique Readability Theorem for propositional formulas (see page 2). To prove the lemma on weights, we assign a weight of  $n - 1$  to each  $n$ -ary function symbol, and -1 to each variable.  $\square$

**Exercise 1** Carry out the details in the above argument.

**Notation:** We use  $r, s, t, \dots$  to denote terms.

In the vocabulary for arithmetic  $\mathcal{L}_A$ , in practice we write  $+, \cdot$  as though they were infix operators, even though officially they are prefix operators. Thus

**Notation**  $(t_1 \cdot t_2) =_{syn} \cdot t_1 t_2$   
 $(t_1 + t_2) =_{syn} + t_1 t_2$

Thus examples of our way of writing  $\mathcal{L}_A$  terms are  $sss0, ((x + sy) \cdot (ssz + s0))$

**Definition of first-order formula in the vocabulary  $\mathcal{L}$**  (or  $\mathcal{L}$ -formula, or just *formula*):

- 1)  $Pt_1 \dots t_n$  is an *atomic*  $\mathcal{L}$ -formula, where  $P$  is an  $n$ -ary predicate symbol in  $\mathcal{L}$  and  $t_1, \dots, t_n$  are  $\mathcal{L}$ -terms.
- 2) If  $A$  and  $B$  are  $\mathcal{L}$ -formulas, so are  $\neg A, (A \wedge B),$  and  $(A \vee B)$
- 3) If  $A$  is an  $\mathcal{L}$ -formula and  $x$  is a variable, then  $\forall x A$  and  $\exists x A$  are  $\mathcal{L}$ -formulas.

As in the case of propositional formulas, we use the notation

$(A \supset B)$  for  $(\neg A \vee B)$

$(A \leftrightarrow B)$  for  $(A \supset B) \wedge (B \supset A)$

Examples of formulas:  $(\neg \forall x Px \vee \exists x \neg Px)$  (Here  $P$  is a unary predicate symbol.)

$(\forall x \neg Qxy \wedge \neg \forall z Qfyz)$ . (Here  $Q$  is a binary predicate symbol and  $f$  is a unary function symbol.)

The Unique Readability Theorem holds for first-order formulas.

**Notation**  $r = s$  stands for  $= rs$

$r \neq s$  stands for  $\neg(r = s)$

**Example:** Goldbach's conjecture: Every even integer greater than 2 is the sum of two primes.

$\forall x((\text{Even}(x) \wedge x > 2) \supset \exists y \exists z(\text{Prime}(y) \wedge \text{Prime}(z) \wedge x = y + z))$

Here Even, Prime are unary predicate symbols.

$>$  is a binary predicate symbol (we use infix notation).

2 is a constant symbol.

$+$  is a binary function symbol.

This can also be stated as a formula in the vocabulary  $\mathcal{L}_A$ , since the predicates Even, Prime, and  $>$  can be defined in terms of  $s, +, \cdot,$  and  $=$ . For example,  $\text{Even}(x)$  can be defined by the formula  $\exists y(x = y + y)$ .

## Free and Bound Variables

**Definition:** An occurrence of  $x$  in  $A$  is *bound* iff it is in a subformula of  $A$  of the form  $\forall xB$  or  $\exists xB$ . Otherwise the occurrence is *free*.

For example, in the formula  $\exists y(x = y + y)$  (which defines  $\text{Even}(x)$  as above) the occurrence of  $x$  is free, while the occurrences of  $y$  are bound. Intuitively the meaning of a formula depends on the values assigned to its free variables, but no value need be assigned to a bound variable to give the formula meaning.

Notice that a variable can have both free and bound occurrences in one formula. For example, in  $Px \wedge \forall xQx$ , the first occurrence of  $x$  is free, and the second occurrence is bound.

**Definition:** A formula  $A$  or a term  $t$  is *closed* if it contains no free occurrence of a variable. A closed formula is called a *sentence*.

## Semantics of Predicate Calculus

In the propositional calculus, a truth assignment provides meaning to a formula. In the predicate calculus, we need a more complicated object, called a *structure* (or *interpretation*)

to give meaning to formulas and terms. If  $\mathcal{L}$  is a first-order vocabulary, then an  $\mathcal{L}$ -structure  $\mathcal{M}$  consists of the following:

- 1) A nonempty set  $M$  called the *universe of discourse* (or just *universe*). Variables in an  $\mathcal{L}$ -formula range over  $M$ .
- 2) For each  $n$ -ary function symbol  $f$  in  $\mathcal{L}$ , an associated function  $f^{\mathcal{M}} : M^n \mapsto M$ .
- 3) For each  $n$ -ary predicate symbol in  $\mathcal{L}$ , an associated relation  $P^{\mathcal{M}} \subseteq M^n$ . If  $\mathcal{L}$  contains  $=$ , then  $=^{\mathcal{M}}$  must be the true equality relation on  $M$ .

Notice that the predicate symbol  $=$  gets special treatment in the above definition, in that  $=^{\mathcal{M}}$  must always be the true equality relation. Other predicate symbols may be interpreted by arbitrary relations of the appropriate arity. For example, if  $\mathcal{L}$  contains the binary predicate symbol  $<$ , then  $<^{\mathcal{M}}$  can be any binary relation on the universe  $M$ , and is not necessarily an order relation.

Every  $\mathcal{L}$ -sentence becomes either true or false when interpreted by an  $\mathcal{L}$ -structure  $\mathcal{M}$ , as explained below. If a sentence  $A$  becomes true under  $\mathcal{M}$ , then we say  $\mathcal{M}$  *satisfies*  $A$ , or  $\mathcal{M}$  is a *model* for  $A$ , and write  $\mathcal{M} \models A$ .

**Definition:** We say that a structure  $\mathcal{M}$  is *finite* if the universe  $M$  of  $\mathcal{M}$  is finite. Otherwise  $\mathcal{M}$  is infinite.

If  $A$  has free variables, then these variables must be interpreted as specific elements in the universe  $M$  before  $A$  gets a truth value under the structure  $\mathcal{M}$ . For this we need the following:

**Definition:** An *object assignment*  $\sigma$  for a structure  $\mathcal{M}$  is a mapping from variables to the universe  $M$ .

Below we give the formal definition of notion  $\mathcal{M} \models \mathcal{A}[\sigma]$ , which is intended to mean that the structure  $\mathcal{M}$  satisfies the formula  $A$  when the free variables of  $A$  are interpreted according to the object assignment  $\sigma$ . First it is necessary to define the notation  $t^{\mathcal{M}}[\sigma]$ , which is the element of universe  $M$  assigned to the term  $t$  by the structure  $\mathcal{M}$  when the variables of  $t$  are interpreted according to  $\sigma$ .

## Basic Semantic Definition

Let  $\mathcal{L}$  be a vocabulary, let  $\mathcal{M}$  be an  $\mathcal{L}$ -structure, and let  $\sigma$  be an object assignment for  $\mathcal{M}$ .

Each  $\mathcal{L}$ -term  $t$  is assigned an element  $t^{\mathcal{M}}[\sigma]$  in  $M$ , defined by structural induction on terms  $t$ , as follows (refer to the definition of  $\mathcal{L}$ -term, page ??):

- a)  $x^{\mathcal{M}}[\sigma]$  is  $\sigma(x)$ , for each variable  $x$
- b)  $(ft_1 \cdots t_n)^{\mathcal{M}}[\sigma] = f^{\mathcal{M}}(t_1^{\mathcal{M}}[\sigma], \dots, t_n^{\mathcal{M}}[\sigma])$

*Notation:* If  $x$  is a variable and  $m \in M$ , then the object assignment  $\sigma(m/x)$  is the same as  $\sigma$  except  $\sigma(m/x)(x) = m$ .

For  $A$  an  $\mathcal{L}$ -formula, the notion  $\mathcal{M} \models A[\sigma]$  ( $\mathcal{M}$  satisfies  $A$  under  $\sigma$ ) is defined by structural induction on formulas  $A$  as follows (refer to the definition of formula):

- a)  $\mathcal{M} \models (Pt_1 \cdots t_n)[\sigma]$  iff  $\langle t_1^{\mathcal{M}}[\sigma], \dots, t_n^{\mathcal{M}}[\sigma] \rangle \in P^{\mathcal{M}}$
- b)  $\mathcal{M} \models (s = t)[\sigma]$  iff  $s^{\mathcal{M}}[\sigma] = t^{\mathcal{M}}[\sigma]$
- c)  $\mathcal{M} \models \neg A[\sigma]$  iff not  $\mathcal{M} \models A[\sigma]$ .
- d)  $\mathcal{M} \models (A \vee B)[\sigma]$  iff  $\mathcal{M} \models A[\sigma]$  or  $\mathcal{M} \models B[\sigma]$ .
- e)  $\mathcal{M} \models (A \wedge B)[\sigma]$  iff  $\mathcal{M} \models A[\sigma]$  and  $\mathcal{M} \models B[\sigma]$ .
- f)  $\mathcal{M} \models (\forall x A)[\sigma]$  iff  $\mathcal{M} \models A[\sigma(m/x)]$  for all  $m \in M$
- g)  $\mathcal{M} \models (\exists x A)[\sigma]$  iff  $\mathcal{M} \models A[\sigma(m/x)]$  for some  $m \in M$

This method of giving meaning is sometimes called Tarski semantics, named after the important logician Alfred Tarski.

Note that item b) in the definition of  $\mathcal{M} \models A[\sigma]$  follows from a) and the fact that  $=^{\mathcal{M}}$  is always the equality relation.

If  $t$  is a closed term (i.e. contains no variables), then  $t^{\mathcal{M}}[\sigma]$  is independent of  $\sigma$ , and so we sometimes just write  $t^{\mathcal{M}}$ . Similarly, if  $A$  is a sentence, then we sometimes write  $\mathcal{M} \models A$  instead of  $\mathcal{M} \models A[\sigma]$ , since  $\sigma$  does not matter. (See the Corollary on the next page.)

**Example:** Let  $\mathcal{L}$  be the vocabulary  $\{; R, =\}$  and let  $\mathcal{M}$  be the  $\mathcal{L}$ -structure whose universe  $M = \mathbb{N}$  and such that  $R^{\mathcal{M}}(m, n)$  holds iff  $m \leq n$ . Then  $\mathcal{M} \models \exists x \forall y R(x, y)$  (since 0 is the least element of  $\mathbb{N}$ ) but  $\mathcal{M} \not\models \exists y \forall x R(x, y)$  since there is no largest natural number.

**Standard Structure:** The *standard structure*  $\underline{\mathbb{N}}$  for the vocabulary  $\mathcal{L}_A$  has universe  $M = \mathbb{N} = \{0, 1, 2, \dots, \}$ ,  $s^{\underline{\mathbb{N}}}(n) = n + 1$ , and  $0, +, \cdot, =$  get their usual meanings on the natural numbers.

**Example:**  $\mathbb{N} \models \forall x \forall y \exists z (x + z = y \vee y + z = x)$  (since either  $y - x$  or  $x - y$  exists) but  $\mathbb{N} \not\models \forall x \exists y (y + y = x)$  since not all natural numbers are even.

In the future we sometimes assume that there is some first-order vocabulary  $\mathcal{L}$  in the background, and do not necessarily mention it explicitly.

**Notation:** In general,  $\Phi$  denotes a set of formulas,  $A, B, C, \dots$  denote formulas,  $\mathcal{M}$  denotes a structure, and  $\sigma$  denotes an object assignment.

**Lemma:** If  $\sigma$  and  $\sigma'$  agree on the free variables of  $A$ , then  $\mathcal{M} \models A[\sigma]$  iff  $\mathcal{M} \models A[\sigma']$ .

**Proof:** Structural induction on formulas  $A$ .

**Corollary:** If  $A$  is a sentence, then for any object assignments  $\sigma, \sigma'$ ,  $\mathcal{M} \models A[\sigma]$  iff  $\mathcal{M} \models A[\sigma']$ .

In view of the Corollary, if  $A$  is a sentence, then  $\sigma$  is irrelevant, so we omit mention of  $\sigma$  and simply write  $\mathcal{M} \models A$ .

**Definition:**

- a)  $A$  is *satisfiable* iff  $\mathcal{M} \models A[\sigma]$  for some  $\mathcal{M}$  and  $\sigma$ .
- b)  $\mathcal{M} \models \Phi[\sigma]$  iff  $\mathcal{M} \models A[\sigma]$  for all  $A \in \Phi$ . (We may omit mention of  $\sigma$  if  $\Phi$  is a set of sentences.) We say  $\Phi$  is *satisfiable* if  $\mathcal{M} \models \Phi[\sigma]$  for some  $\mathcal{M}$  and  $\sigma$ .
- c)  $\Phi \models A$  iff for all  $\mathcal{M}$  and all  $\sigma$ , if  $\mathcal{M} \models \Phi[\sigma]$  then  $\mathcal{M} \models A[\sigma]$ .
- d)  $\models A$  ( $A$  is *valid*) iff  $\mathcal{M} \models A[\sigma]$  for all  $\mathcal{M}$  and  $\sigma$ .
- e)  $A \iff B$  ( $A$  and  $B$  are *logically equivalent*, or just *equivalent*) iff for all  $\mathcal{M}$  and all  $\sigma$ ,  $\mathcal{M} \models A[\sigma]$  iff  $\mathcal{M} \models B[\sigma]$ .

$\Phi \models A$  is read “ $A$  is a logical consequence of  $\Phi$ ”. This relation is of FUNDAMENTAL IMPORTANCE. Do not confuse this with our other use of the symbol  $\models$ , as in  $\mathcal{M} \models A$  ( $\mathcal{M}$  satisfies  $A$ ). In the latter,  $\mathcal{M}$  is a structure, rather than a set of formulas.

Note that  $\models$  is a symbol of the “meta language” (English), as opposed to  $\neg, \vee, \wedge, \forall, \exists$ , which are symbols of the “object language”.

As in the propositional case, if  $\Phi = \{B_1, \dots, B_n\}$ , then we sometimes write  $B_1, \dots, B_n \models A$  instead of  $\{B_1, \dots, B_n\} \models A$ .

**Examples:**

- 1  $(\forall x A \vee \forall x B) \models \forall x (A \vee B)$ , for all formulas  $A$  and  $B$ .

Proof: We follow the definition of  $\Phi \models A$  above. Let  $\mathcal{M}$  be any structure and let  $\sigma$  be any object assignment. Assume L.H.S. is true, i.e.  $\mathcal{M} \models (\forall x A \vee \forall x B)[\sigma]$ .

Then following the Basic Semantic Definition,  $\mathcal{M} \models (\forall xA)[\sigma]$  or  $\mathcal{M} \models (\forall xB)[\sigma]$ . Say  $\mathcal{M} \models (\forall xA)[\sigma]$ . Then  $\mathcal{M} \models A[\sigma(m/x)]$  for all  $m \in M$ . Then  $\mathcal{M} \models (A \vee B)[\sigma(m/x)]$  for all  $m \in M$ . Therefore  $\mathcal{M} \models \forall x(A \vee B)[\sigma]$ .

Similarly for the case  $\mathcal{M} \models (\forall xB)[\sigma]$ .  $\square$

2  $\forall x(A \vee B) \models (\forall xA \vee \forall xB)$ ? No, not necessarily.

Take  $A =_{syn} Px$ ,  $B =_{syn} Qx$ , define the structure  $\mathcal{M}$  to have universe  $M = \mathbb{N}$ , define  $P^{\mathcal{M}}$  to be the set of even natural numbers, and  $Q^{\mathcal{M}}$  to be the set of odd natural numbers. Then  $\mathcal{M} \models \forall x(Px \vee Qx)$  (every number is even or odd), but not  $\mathcal{M} \models (\forall xPx \vee \forall xQx)$  (it is not the case that either all numbers are even or all numbers are odd).

3  $\neg \forall xA \iff \exists x \neg A$ , for all formulas  $A$ .

$\neg \exists xA \iff \forall x \neg A$ , for all formulas  $A$ .

$(\forall xA \wedge \forall xB) \iff \forall x(A \wedge B)$ , for all formulas  $A, B$ .

$\exists x(A \vee B) \iff (\exists xA \vee \exists xB)$ , for all formulas  $A, B$ .

$\exists x(A \wedge B) \models (\exists xA \wedge \exists xB)$ , for all formulas  $A, B$ .

NOT  $(\exists xA \wedge \exists xB) \models \exists x(A \wedge B)$  in general

$\forall x \forall y A \iff \forall y \forall x A$

$\exists x \exists y A \iff \exists y \exists x A$

$\exists y \forall x A \models \forall x \exists y A$ , for all formulas  $A$ .

NOT  $\forall x \exists y A \models \exists y \forall x A$  in general

$\forall x A \models \exists x A$ , because of our requirement that every universe  $M$  must be nonempty.

$\forall x \forall y (x = y \supset fx = fy)$  is valid.

$\forall x \forall y (fx = fy \supset x = y)$  is NOT valid.

**Exercise 2** Verify each line in item 3 above. For the two lines beginning NOT give specific formulas  $A$  (and  $B$ ) for which the relation is false, and show it is false by giving a specific structure which satisfies the left hand side but not the right hand side. For the last line, give a structure which does not satisfy the formula.

**Exercise 3** Show that  $\{P0, Ps0, Pss0, \dots\} \not\models \forall x Px$  by giving a specific structure.

**Exercise 4** Consider the following four formulas over the vocabulary  $\mathcal{L}_A$ :

$P1: \forall x (sx \neq 0)$

$P2: \forall x \forall y (sx = sy \supset x = y)$

$P3: \forall x (x + 0 = x)$

$P4: \forall x \forall y (x + sy = s(x + y))$

Prove from the definition of  $\models$  that

$$P1, P2, P3, P4 \not\models \forall x \forall y (x + y = y + x)$$

**Hint:** Think of  $+$  as string concatenation.

**Exercise 5** Show that  $\forall x (g f x = x)$  is NOT a logical consequence of  $\forall x (f g x = x)$ .

**Exercise 6** Let  $\mathcal{M}$  be a structure and let  $\Phi$  be the set of all sentences  $A$  satisfied by  $\mathcal{M}$ . Show that  $\Phi$  is closed under  $\models$ . That is, show that if  $\Phi \models A$  then  $A \in \Phi$ .

**Exercise 7** Give a sentence in the vocabulary  $\mathcal{L} = \{; =\}$  which is satisfied by a structure iff the universe has exactly three elements.

**Exercise 8** Give a satisfiable sentence  $A$  in the vocabulary  $\mathcal{L} = \{; R\}$ , where  $R$  is a binary predicate symbol, such that  $A$  has no finite model. (Hint: Think of  $R$  as an order relation.)

**Exercise 9** Give a sentence  $A$  in the vocabulary  $\mathcal{L} = \{; R, =\}$ , where  $R$  is a binary predicate symbol, such that for all  $n \in \mathbb{N}, n > 0$ ,  $A$  has a model whose universe has  $n$  elements iff  $n$  is even. (Hint: Think of  $R$  as a pairing relation.)

**Exercise 10** Give a sentence  $A$  of the predicate calculus with the vocabulary  $\mathcal{L} = \{; R, =\}$ , where  $R$  is a binary predicate symbol, such that a finite  $\mathcal{L}$ -structure (thought of as a directed graph with edge relation  $R$ ) is a model for  $A$  iff it is a disjoint union of directed cycles. Now give an infinite model for  $A$ .

Recall that a sentence is a formula with no free variables. Each sentence in the vocabulary  $\mathcal{L}_{\mathcal{A}}$  (the vocabulary of arithmetic) is either true or false in the standard structure  $\mathbb{N}$ . Thus  $\forall x \forall y (x + y = y + x)$  and Fermat's Last Theorem are true, while  $\forall x \neg (0 = x + x)$  is false, and no one knows the truth value of Goldbach's conjecture. On the other hand, a formula such as  $\forall y \neg (x = y + y)$  (" $x$  is odd") has no truth value under any structure, since it has a free variable. Of course it gets a truth value in a structure when an object assignment  $\sigma$  is specified.

## Substitution

**Syntactic Definition:** ( $s, t$  are terms)

$t(s/x)$  is the result of replacing all occurrences of  $x$  in  $t$  by  $s$ .

$A(s/x)$  is the result of replacing all *free* occurrences of  $x$  in  $A$  by  $s$ .

**Semantics:**



**Lemma** For each structure  $\mathcal{M}$  and each object assignment  $\sigma$ ,

$$(t(s/x))^{\mathcal{M}}[\sigma] = t^{\mathcal{M}}[\sigma(m/x)]$$

where  $m = s^{\mathcal{M}}[\sigma]$ .

**Example:** Let  $\mathcal{M}$  be the standard structure  $\mathbb{N}$  for the vocabulary  $\mathcal{L}_A$  of arithmetic. Suppose  $\sigma(x) = 5$  and  $\sigma(y) = 7$ . Let  $s$  be the term  $x + y$  and let  $t$  be the term  $ss0$  (here  $s$  is the successor function in  $\mathcal{L}$ ). Then  $s(t/x)$  is  $ss0 + y$  and so  $(s(t/x))^{\mathbb{N}}[\sigma] = 2 + 7 = 9$ . On the other hand,  $m = t^{\mathbb{N}} = 2$ , so  $s^{\mathbb{N}}[\sigma(m/x)] = 2 + 7 = 9$ , and the Lemma is verified for this case.

**Proof if the Lemma:** Structural induction on  $t$ .

Base case:  $t$  is a variable. If the variable is  $x$ , then both sides of the equation are the same, namely  $s^{\mathcal{M}}[\sigma]$ . If  $t$  is a variable  $y$  other than  $x$ , then again both sides are the same, namely  $\sigma(y)$ .

The induction step is straightforward from the Basic Semantic Definition.  $\square$

**Exercise 11** Carry out the induction step in detail.

Question: Does the above lemma apply to formulas  $A$ ? I.e. can we say  $\mathcal{M} \models A(t/x)[\sigma]$  iff  $\mathcal{M} \models A[\sigma(m/x)]$ , where  $m = t^{\mathcal{M}}[\sigma]$ ? Something can go wrong.

Example: Suppose  $A$  is  $\forall y \neg(x = y + y)$ . This says “ $x$  is odd”. But  $A(x + y/x)$  is  $\forall y \neg(x + y = y + y)$ , which does not say “ $x + y$  is odd” as desired, but instead it is always false. The problem is that  $y$  in the term  $x + y$  got “caught” by the quantifier  $\forall y$ .

**Definition** A term  $t$  is *freely substitutable for  $x$  in  $A$*  iff no free occurrence of  $x$  in  $A$  is in a subformula of  $A$  of the form  $\forall y B$  or  $\exists y B$ , where  $y$  occurs in  $t$ .

**Substitution Theorem:** If  $t$  is freely substitutable for  $x$  in  $A$  then for all structures  $\mathcal{M}$  and all object assignments  $\sigma$ ,  $\mathcal{M} \models A(t/x)[\sigma]$  iff  $\mathcal{M} \models A[\sigma(m/x)]$ , where  $m = t^{\mathcal{M}}[\sigma]$ .

**Proof:** Structural induction on  $A$ . The interesting case is when  $A$  is  $\forall y B$ . (The case when  $A$  is  $\exists y B$  is similar). Then we are to prove

$$\mathcal{M} \models (\forall y B)(t/x)[\sigma] \text{ iff } \mathcal{M} \models (\forall y B)[\sigma(m/x)] \quad (1)$$

where  $m = t^{\mathcal{M}}[\sigma]$ .

If  $x$  does not occur free in  $\forall y B$ , then no substitution is done, so the result is easy. (If  $x, y$  are the same variable, then  $x$  does not occur free in  $\forall y B$ .)

Hence we may assume that  $x, y$  are distinct variables and  $x$  occurs free in  $B$ . Since  $t$  is freely substitutable for  $x$  in  $\forall y B$ ,  $y$  does not occur in  $t$ .

Following the Basic Semantic Definition, the LHS of (??) holds iff  $\mathcal{M} \models B(t/x)[\sigma(n/y)]$  for all  $n \in M$ . Apply the induction hypothesis to  $B$  to obtain

$$\mathcal{M} \models B(t/x)[\sigma(n/y)] \text{ iff } \mathcal{M} \models B[\sigma(n/y)(m'/x)]$$

where now  $m' = t^{\mathcal{M}}[\sigma(n/y)]$ . But note that  $m' = t^{\mathcal{M}}[\sigma(n/y)] = t^{\mathcal{M}}[\sigma] = m$  because  $y$  does not occur in  $t$ . Hence

$$\mathcal{M} \models B(t/x)[\sigma(n/y)] \text{ iff } \mathcal{M} \models B[\sigma(n/y)](m/x)$$

Now the RHS of (??) holds iff  $\mathcal{M} \models B[\sigma(m/x)](n/y)$  for all  $n \in M$ . But  $\sigma(n/y)(m/x) = \sigma(m/x)(n/y)$ , since  $x$  and  $y$  are distinct. Hence the LHS holds iff the RHS holds.  $\square$

### Change of Bound Variable

If a term  $t$  is not freely substitutable for  $x$  in  $A$ , it is because some variable  $y$  in  $t$  gets caught by a quantifier  $\forall y$  or  $\exists y$  in  $A$ . One way to fix this is simply rename the bound variable  $y$  in  $A$  to some new variable  $z$ . It should be intuitively clear that this renaming does not change the meaning of  $A$ . The definition and lemmas below formalize this process.

**Definition:**  $\forall zA(z/y)$  results from  $\forall yA$  by *change of bound variable* provided  $z$  does not occur in  $A$ . Similarly for  $\exists zA(z/y)$ .

**Lemma:** If  $z$  does not occur in  $A$ , then  $\forall zA(z/y)$  and  $\forall yA$  are logically equivalent. Also  $\exists zA(z/y)$  and  $\exists yA$  are equivalent.

**Proof:** This follows from the Basic Semantic Definition and the Substitution Theorem. (Verify this).  $\square$

**Definition**  $A'$  is a *variant* of  $A$  if  $A'$  results by a sequence of changes of bound variables to subformulas of  $A$ .

**Theorem:** If  $A'$  is a variant of  $A$  then  $A$  and  $A'$  are equivalent.

This follows from the preceding Lemma and the following general result:

**Replacement Theorem:** If  $B$  and  $B'$  are equivalent formulas and  $A'$  results from  $A$  by replacing some occurrence of  $B$  in  $A$  by  $B'$ , then  $A$  and  $A'$  are equivalent.

**Exercise 12** Prove the Replacement Theorem, by structural induction on  $A$  (relative to  $B$ ). The base case is when  $A$  and  $B$  coincide.

**Example:**  $B$  is  $\neg\forall xPxy$ ,  $B'$  is  $\exists z\neg Pzy$ ,  $A$  is  $\forall y(\neg\forall xPxy \supset Qy)$ . Note that  $B$  has a free variable that is bound in  $A$ .  $A'$  is  $\forall y(\exists z\neg Pzy \supset Qy)$ . By the Replacement Theorem,  $A$  and  $A'$  are equivalent, even though the quantifier  $\forall y$  in  $A$  catches a variable in  $B$ .

## A First-Order Gentzen System

We now extend the propositional proof system  $PK$  to the first-order sequent proof system  $LK$ . For this it is convenient to introduce two kinds of variables:

- type “free”:  $a, b, c, \dots$

- type “bound”:  $x, y, z, \dots$

A first-order formula  $A$  is called a *proper formula* if it satisfies the restriction that every variable that occurs free has type free, and every variable that occurs bound has type bound. Similarly a *proper term* has no variable of type bound. Notice that a subformula of a proper formula is not necessarily proper, and a proper formula may contain terms which are not proper.

The sequent system  $LK$  is an extension of the propositional system  $PK$ , where now all formulas  $A_1, \dots, A_k, B_1, \dots, B_\ell$  in a sequent  $A_1, \dots, A_k \rightarrow B_1, \dots, B_\ell$  must be proper formulas. In addition to the rules given for  $PK$ , the system  $LK$  has four rules for introducing the quantifiers.

**Notation:** In the rules below,  $t$  is any proper term and  $A(t)$  is the result of substituting  $t$  for all free occurrences of  $x$  in  $A(x)$ . Similarly  $A(b)$  is the result of substituting  $b$  for all free occurrences of  $x$  in  $A(x)$ . Note that  $t$  and  $b$  can always be freely substituted for  $x$  in  $A(x)$  because  $\forall xA(x)$  and  $\exists xA(x)$  are proper formulas.

$\forall$  introduction rules

$$\text{left } \frac{A(t), \Gamma \rightarrow \Delta}{\forall xA(x), \Gamma \rightarrow \Delta} \qquad \text{right } \frac{\Gamma \rightarrow \Delta, A(b)}{\Gamma \rightarrow \Delta, \forall xA(x)}$$

$\exists$  introduction rules

$$\text{left } \frac{A(b), \Gamma \rightarrow \Delta}{\exists xA(x), \Gamma \rightarrow \Delta} \qquad \text{right } \frac{\Gamma \rightarrow \Delta, A(t)}{\Gamma \rightarrow \Delta, \exists xA(x)}$$

**Restriction:** The free variable  $b$  must not occur in the conclusion in  $\forall$  **right** and  $\exists$  **left**.

**Example:** An instance of  $\forall$ -**left** is

$$\frac{Pbb \rightarrow Pbb}{\forall yPby \rightarrow Pbb}$$

What is the formula  $A(y)$  in this case?

### Semantics of first-order sequents

The semantics of first-order sequents is a natural generalization of the semantics of propositional sequents given on page 10. Again a sequent  $S =_{syn}$

$$A_1, \dots, A_k \rightarrow B_1, \dots, B_\ell$$

has the same meaning as its associated formula  $A_S =_{syn}$

$$(A_1 \wedge A_2 \wedge \dots \wedge A_k) \supset (B_1 \vee B_2 \vee \dots \vee B_\ell) \tag{2}$$

In particular, we say that the sequent is *valid* iff its associated formula is valid.

**Definition:** [Universal Closure] Suppose that  $A$  is a formula whose free variables comprise the list  $a_1, \dots, a_n$ . Then the *universal closure* of  $A$ , written  $\forall A$ , is the sentence  $\forall x_1 \dots \forall x_n A(x_1/a_1, \dots, x_n/a_n)$ , where  $x_1, \dots, x_n$  are new (bound) variables. If  $\Phi$  is a set of formulas, then  $\forall\Phi$  is the set of all sentences  $\forall A$ , for  $A$  in  $\Phi$ .

Note that every formula  $A$  is valid iff its universal closure  $\forall A$  is valid. Also  $A$  is a logical consequence of its universal closure  $\forall A$ , but  $\forall A$  is not necessarily a logical consequence of  $A$  (for example take  $A =_{syn} Pa$ ).

Recall that for the propositional system  $PK$ , for each rule the bottom sequent is a logical consequence of the top sequent(s). This remains true for  $LK$ , with the exception of the rules  $\forall$ -**right** and  $\exists$ -**left**. For these rules we can make a weaker statement: the universal closure of (the meaning of) the bottom sequent is a logical consequence of the universal closure of (the meaning of) the top sequent. The following proposition makes this weaker statement for all the  $PK$  rules. (The statement is weaker, because for any formulas  $A$  and  $B$ , if  $A \models B$ , then  $\forall A \models \forall B$ ).

**Lemma** For each  $PK$  rule, the universal closure of the meaning of the bottom sequent is a logical consequence of the universal closure(s) of the meaning(s) of the top sequent(s). Here the *meaning* of a sequent  $S$  is the formula  $A_S$  given in (??).

**Proof:** The argument for the propositional rules is essentially the same as for the system  $PK$ . The arguments for  $\forall$ -**left** and  $\exists$ -**right** are easy; and in fact in these cases it is not necessary to take universal closures.

We illustrate the remaining two rules by considering the case of  $\forall$ -**right**. Note that because of the **Restriction** for this rule, the variable  $b$  cannot occur in  $\Gamma$  or  $\Delta$ . Hence it suffices to verify that

$$\forall x(\bigwedge \Gamma \supset (\bigvee \Delta \vee A(x))) \models \bigwedge \Gamma \supset (\bigvee \Delta \vee \forall x A(x))$$

To see that this logical consequence holds, suppose that  $\mathcal{M}$  is a structure and  $\sigma$  is an object assignment. Suppose that  $\mathcal{M}$  satisfies the left hand side under  $\sigma$ , i.e.

$$\mathcal{M} \models \forall x(\bigwedge \Gamma \supset (\bigvee \Delta \vee A(x)))[\sigma]$$

Either  $\mathcal{M}$  satisfies  $\forall x A(x)$  under  $\sigma$  or not. In the first case it follows immediately that  $\mathcal{M}$  satisfies the right hand side under  $\sigma$ . In the second case, it must be that

$$\mathcal{M} \models \forall x(\bigwedge \Gamma \supset \bigvee \Delta)[\sigma]$$

and hence again  $\mathcal{M}$  satisfies the right hand side under  $\sigma$ . □

**Exercise 13** Give the argument for the other three quantifier rules.

**Soundness Theorem for  $LK$ :** Every sequent provable in  $LK$  is valid.

**Proof:** This is proved by induction on the number of sequents in the  $LK$  proof. For the base case, obviously each axiom  $A \rightarrow A$  is valid. For the induction step, it follows from the

above lemma that for each rule, if all sequents on top are valid, then the sequent on the bottom is valid.  $\square$

**Exercise 14** Give a specific example of a sequent  $\Gamma \rightarrow \Delta, A(b)$  which is valid, but the bottom sequent  $\Gamma \rightarrow \Delta, \forall xA(x)$  is not valid, because the restriction for the  $\forall$  **right** rule is violated (i.e.  $b$  occurs in  $\Gamma$  or  $\Delta$  or  $\forall xA(x)$ ). Do the same for the  $\exists$  **left** rule.

An *LK* proof of a valid first-order sequent can be obtained using the same method as in the propositional case: Write the goal sequent at the bottom, and move up by using the introduction rules in reverse. A good heuristic is: if there is a choice about which quantifier to remove next, choose  $\forall$  **right** and  $\exists$  **left** first (working backwards), since these rules carry a restriction.

Here is an *LK* proof of the sequent  $(\forall xPx \vee \forall xQx) \rightarrow \forall x(Px \vee Qx)$ .

$$\frac{\frac{\frac{Pb \rightarrow Pb}{Pb \rightarrow Pb, Qb} \text{ (weakening)}}{Pb \rightarrow (Pb \vee Qb)} \text{ (\vee right)}}{\forall xPx \rightarrow (Pb \vee Qb)} \text{ (\forall left)} \quad \frac{\frac{\frac{Qb \rightarrow Qb}{Qb \rightarrow Pb, Qb} \text{ (weakening)}}{Qb \rightarrow (Pb \vee Qb)} \text{ (\vee right)}}{\forall xQx \rightarrow (Pb \vee Qb)} \text{ (\forall left)}}{\frac{(\forall xPx \vee \forall xQx) \rightarrow (Pb \vee Qb)}{(\forall xPx \vee \forall xQx) \rightarrow \forall x(Px \vee Qx)} \text{ (\forall right)}}$$

**Exercise 15** Give *LK* proofs for the following valid sequents:

$$\begin{aligned} \forall xPx \wedge \forall xQx &\rightarrow \forall x(Px \wedge Qx) \\ \forall x(Px \wedge Qx) &\rightarrow \forall xPx \wedge \forall xQx \\ \exists x(Px \vee Qx) &\rightarrow \exists xPx \vee \exists xQx \\ \exists xPx \vee \exists xQx &\rightarrow \exists x(Px \vee Qx) \\ \exists x(Px \wedge Qx) &\rightarrow \exists xPx \wedge \exists xQx \\ \exists y\forall xPxy &\rightarrow \forall x\exists yPxy \\ \forall xPx &\rightarrow \exists xPx \end{aligned}$$

Check that the rule restrictions seem to prevent generating *LK* proofs for the following invalid sequents:

$$\begin{aligned} \exists xPx \wedge \exists xQx &\rightarrow \exists x(Px \wedge Qx) \\ \forall x\exists yPxy &\rightarrow \exists y\forall xPxy \end{aligned}$$