Basic Logic 4

Frank Veltman
ILLC
University of Amsterdam

2009

1 First Order Predicate Logic

1.1 Grammar

A language $\mathcal{L}$ of first order predicate logic is given by six sets of symbols, containing the (individual) *constants*, the (individual) *variables*, the *predicates*, the *function symbols*, the *logical constants* and the *punctuation marks* of $\mathcal{L}$.

Individual constants Individual constants are the names of the objects in the domain of discussion. In the metalanguage the characters a, b, c — possibly with natural numbers as index — are used to refer to individual constants..

Individual variables There are countably many individual variables is countably infinite. The characters u, v, w, x, y and z will be used to refer to them, sometimes with natural numbers as indices, like so: $x_1, y_1, z_1, x_2, y_2, \ldots$.

Predicates Each predicate comes with a natural number. This number indicates the *arity*, i.e. the number of argument places of the predicate. As we will see when we get to the semantics, 1-place predicates express properties, 2-place stand for relations. A language can have 0-place predicates; these function as (atomic) sentences. $\bot$ is a special 0-place predicate that is also an element of the set of logical constants. A language of predicate logic often has a two-place predicate standing for the identity relation. I will refer to this predicate with the sign "=", and treat it as a logical constant. To the other predicates of a language $\mathcal{L}$ we will refer with capital characters like P, Q and R, using the character P mostly to refer to a one-place predicate, and the character R to refer to a two-place predicate.

Function symbols You may not have seen function symbols before. Usually they are not treated in a first logic course. Because of the importance of functions in scientific discourse — think of the addition operator "+" in arithmetic or of the mass function in mechanics, etc. — we include them in our languages here. I will often use the characters f, g and h to refer to function symbols. Just like predicates function symbols have some arity, given by a natural numbers. Individual constants can be thought of as 0-place function symbols. [1]

Logical constants In addition to the logical constants that you know from propositional logic, the languages of first order predicate logic have a *universal quantifier* "∀", and an *existential quantifier* "∃". As already indicated we will also treat the identity sign "=" as a logical constant.

Punctuation marks You already know the *left bracket* "(" and the *right bracket* ")". To these, we add the *comma* ",".

So much for the symbols of the languages of first order predicate logic. By concatenating the symbols of a language $\mathcal{L}$ one forms *expressions* of $\mathcal{L}$. Not all expressions of $\mathcal{L}$ are well-formed. Only the *terms* and the *formulas* are.

Definition 1 (Term, Formula)
Let $\mathcal{L}$ be a language of first order predicate logic. The set of terms *of $\mathcal{L}$ is the smallest set X of expressions $\mathcal{L}$ meeting the following conditions.*

(i) If a is a individual constant (of $\mathcal{L}$), then $a \in X$.

(ii) If x is a variabele, then $x \in X$.

(iii) If f is a n-ary function symbol, and $t_1, \ldots, t_n \in X$, then $f(t_1, \ldots, t_n) \in X$.[2]

The set of formulas *of $\mathcal{L}$ is the smallest set X of expressions of $\mathcal{L}$ meeting the following conditions.*

(i) If Q is a n-ary predicate, and $t_1, \ldots, t_n$ are terms, then $Q(t_1, \ldots, t_n) \in X$.[3,4,5]

[1] See footnote 4 for an explanation.

[2] Read "$f(t_1, \ldots, t_n)$" as "the concatenation of f, the left bracket, t_1, the comma, t_2, the comma, etc. until t_n, and finally the right bracket."

[3] We will write '$(t_1 = t_2)$' instead of '$= (t_1, t_2)$'.

[4] Notice that an n-ary function symbol needs n terms to form a new term, and an n-ary predicate needs n terms to form a sentence. Looking at it this way, a 0-ary term is something that needs zero terms to form a new term, and a 0-place predicate is something that needs zero terms to form a sentence. Individual constants satisfy the first condition, and atomic sentences like ⊥ the second.

[5] We will often omit the brackets and comma's in expressions like $P(t_1, \ldots, t_n)$ and just write $Pt_1 \ldots t_n$.

(ii) If $\varphi, \psi \in X$, then $\neg\varphi, (\varphi \wedge \psi), (\varphi \vee \psi), (\varphi \rightarrow \psi) \in X$.

(iii) If $\varphi \in X$, and x is a variable, then $\forall x\varphi \in X$ and $\exists x\varphi \in X$.

Definition 2 (Free and bound variables)
Let $\mathcal{L}$ be a language of first order predicate logic. Let x be a variable of $\mathcal{L}$ and φ a formula of $\mathcal{L}$. The variable x occurs bound *at position k in φ iff:*

(i) x occurs at position k in φ.

(ii) Neither the quantifier $\exists$, nor the quantifier $\forall$ occurs at position $k-1$ in φ.

(iii) There is some formula of the form $\forall x\psi$ or $\exists x\psi$ occurring in φ at position $k-j$ until $k+i$ (for some $i, j \geq 0$).

The variable x occurs free *at position k in φ iff:*

(i) x occurs at position k in φ.

(ii) Neither $\forall$, nor $\exists$ occur at position $k-1$ in φ.

(iii) There is no formula of the form $\forall x\psi$ or $\exists x\psi$ that occurs in φ at place $k-j$ until $k+i$ for some $i, j \geq 0$.

Definition 3 (Sentence, Theory)
Let $\mathcal{L}$ be a language of first order predicate logic. A formula φ of $\mathcal{L}$ is a sentence *of $\mathcal{L}$ iff no variable of $\mathcal{L}$ occurs free somewhere in φ. A* theory *in $\mathcal{L}$ is a set of sentences of $\mathcal{L}$.*

Examples of Theories

Orderings
Let $\mathcal{L}$ be a language of first order predicate logic with one two-place predicate '$<$'. Instead of '$<(x, y)$' we will write '$x < y$'. The next theory is known as the theory of (strict) *partial order.*

1. $\forall x \neg x < x$ irreflexivity.
2. $\forall x \forall y \forall z((x < y \wedge y < z) \rightarrow x < z)$ transitivity.

One gets the theory of *linear order* by adding the next sentence

3. $\forall x \forall y(x \neq y \rightarrow (x < y \vee y < x))$ connectedness.

A *dense* linear ordering is een linear ordering for which the following holds:

4. $\forall x \forall y(x < y \rightarrow \exists z(x < z \wedge z < y))$ density.

And the theory of dense linear orders without end points is given by the sentences 1, 2, 3, 4 plus:

5. $\forall x \exists y (x < y)$ no endpoint.
6. $\forall x \exists y (y < x)$ no beginning.

Question: A linear order is often pictured as a number of points lying on a straight line. Why are we allowed to do so?

Peano's arithmetic
This theory is formulated in a language with the following constants:

- The individual constant 0.
- The 1-place function symbol S. (Read '$S(x)$' as 'the successor of x'. We will write 'Sx' rather than'$S(x)$'.)
- The 2-place function symbols $+$ and $\times$. We will write '$(x + y)$' en '$(x \times y)$' instead of '$+(x, y)$' and '$\times(x, y)$'.

The axioms of Peano's arithmetic are:

$\forall x (Sx \neq 0)$

$\forall x (Sx \neq x)$

$\forall x \forall y (Sx = Sy \rightarrow x = y)$

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

$\forall x \forall y ((x + Sy) = S(x + y))$

$\forall x ((x \times 0) = 0)$

$\forall x \forall y ((x \times Sy) = (x \times y) + x)$

Plus all sentences of the form:

$$([0/x]\varphi \wedge \forall x(\varphi \rightarrow [Sx/x]\varphi)) \rightarrow \forall x \varphi$$

This formula expresses the principle of induction

1.2 Natural deduction rules

One more definition before we turn to the introduction and elimination rules of the quantifiers.

Definition 4 (Substitution of a term for a variable) *Let φ be a formula of $\mathcal{L}$, x a variable, and t a term. We define $[t/x]\varphi$ to be the formula φ' that one gets from φ by substituting t for x at every position where x occurs free in φ. The term t is* free for *x* in *φ iff none of the variables occuring in t get bound when t is substituted for x in φ.*

Examples

i) $[y/x]\forall xPx = \forall xPx$

ii) $[x/y]\exists xRxy = \exists xRxx$

iii) $[x/y]\exists zRzy = \exists zRzx$

iv) $[y/x](\forall xPx \rightarrow Px) = \forall xPx \rightarrow Py$

x is not free for y in $\exists xRxy$, y is free for x in Ryz.

Elimination rule for the universal quantifier

$$\begin{array}{lll} & \vdots & \\ m. & \forall x\varphi & \\ & \vdots & \\ n. & [t/x]\varphi & E_{\forall}, m \end{array}$$

Requirement:

1. t is free for x in φ.

Exercise 1 *Give an example showing that things can go terribly wrong without this restriction on the application of $E_{\forall}$.*

Introduction rule for the universal quantifier

$$\begin{array}{lll} & \vdots & \\ m. & [y/x]\varphi & \\ & \vdots & \\ n. & \forall x\varphi & I_{\forall}, m \end{array}$$

Requirements

1. y is free for x in φ.
2. y does not occur free in $\forall x\varphi$.
3. y does not occur free in a premise or in an assumption that is active at line m.

Of course, the introduction of a universal quantifier is only justified if *all* objects in the domain of discussion satisfy the formula φ. But this idea cannot serve as a basis for the introduction rule for the universal quantifier, simply because it not always feasible to prove for each and every object in the domain whether it satisfies φ. In particular in the case of an infinite domain this is problematic. Therefore, the introduction rule is based on a slightly different idea. It says that generalization over a given variable — the variable y in the scheme above — is admitted, provided that this variable can be thought of as refering to an *arbitrary* object in the domain of discussion. The restrictions on the rule guarantee that this is the case.

Exercise 2 *Give an example showing that without the restrictions 2 and 3 one can derive things that are 'counterintuitive'.*

Introduction rule for the existential quantifier

The rule $I_\exists$ is the mirror image of $E_\forall$. Schematically it looks like this:

$$\begin{array}{lcl} & \vdots & \\ m. & [t/x]\varphi & \\ & \vdots & \\ n. & \exists x\varphi & I_\exists, m \end{array}$$

Restrictions:

1. t is free for x in φ.

Elimination rule for the existential quantifier

$$\begin{array}{lcl} & \vdots & \\ k. & \exists x\varphi & \\ & \vdots & \\ m. & [y/x]\varphi \to \psi & \\ & \vdots & \\ n. & \psi & E_\exists, k, m \end{array}$$

Restrictions:

1. y is free for x in φ.
2. y does not occur free in $\exists x\varphi$.
3. y does not occur free in a premise or in an assumption that is active at line n.
4. y does not occur free in ψ.

Here, too, the variable y is supposed to refer to an arbitrary object. All we can assume about this object is that it has the property expressed by φ. Roughly put the argument runs as follows: "There is some object satisfying the formula φ ... whichever object this is, ψ holds ... therefore ψ holds".

Exercise 3 *Show that dat condition 4 is indispensible.*

Exercise 4 *Give derivations showing that:*

(a) $\forall x Axx \vdash Aaa$

(b) $\forall x \forall y Axy \vdash Aab$

(c) $\forall x \forall y Axy \vdash Aaa$

(d) $\forall x(Ax \wedge Bx) \vdash \forall x Ax \wedge \forall x Bx$

(e) $\forall x Ax \wedge \forall x Bx \vdash \forall x(Ax \wedge Bx)$

(f) $\forall x(Ax \rightarrow Bx), \forall x Ax \vdash \forall x Bx$

(g) $\neg \exists x Ax \vdash \forall x \neg Ax$

(h) $\neg \exists x \neg Ax \vdash \forall x Ax$

(i) $\exists x(Ax \wedge Bx) \vdash \exists x Ax \wedge \exists x Bx$

(j) $\forall x(Ax \rightarrow Bx), \exists x Ax \vdash \exists x Bx$

(k) $\exists x \neg Ax \vdash \neg \forall x Ax$

(l) $\forall x \neg Ax \vdash \neg \exists x Ax$

(m) $\neg \forall x Ax \vdash \exists x \neg Ax$

Exercise 5 *Give derivations showing that:*

(a) $\forall x \forall y(Rxy \rightarrow \neg Ryx) \vdash \forall x \neg Rxx$

(b) $\forall x \exists y Rxy$, $\forall x \forall y(Rxy \rightarrow Ryx)$, $\forall x \forall y \forall z((Rxy \wedge Ryz) \rightarrow Rxz) \vdash \forall x Rxx$

(c) $\neg \exists x((Sx \wedge Mx) \wedge Px)$, $\forall x(Sx \rightarrow Mx)$, $\forall x(Mx \rightarrow Px) \vdash \neg \exists x Sx$

(d) $\neg \exists x Ax \vee \neg \exists x Bx, \exists x(Cx \wedge Ax) \vdash \exists x(Cx \wedge \neg Bx)$

(e) $\exists x(Fx \rightarrow Gxx)$, $\forall x(Fx \wedge Hx)$, $\forall x(\neg Hx \vee \neg \exists y Gyx) \vdash \bot$

Exercise 6
Prove the following: If $\Delta \vdash [a/x]\varphi$ for an individual constant a that occurs nowhere in φ or in the formulas $\psi \in \Delta$, then $\Delta \vdash \forall x \varphi$.

1.3 Rules for identity

In this section we will investigate how our natural deduction system can be extended if in addition to $\forall, \exists, \rightarrow, \wedge, \vee$ and $\neg$ also the identity sign $=$ is treated as a logical constant.

Introduction rule for identity:

$$\begin{array}{lll} & \vdots & \\ n. & t = t & I_= \end{array}$$

This rule says that we can introduce a formula of the form $t = t$ at every step n in a derivation, whichever term t is.

Elimination rule for the identity sign

$$\begin{array}{lll} & \vdots & \\ k. & s = t & \\ & \vdots & \\ m. & [s/x]\varphi & \\ & \vdots & \\ n. & [t/x]\varphi & E_=, k, m \end{array}$$

Restrictions:

1. s and t are free for x in φ.

In words: if at step k in a derivation you have proved that s is identical to t, and at step m you have concluded that $[s/x]\varphi$, which is to say that s has the property expressed by φ, then it is valid to conclude that t has the property expressed by φ as well: $[t/x]\varphi$.

Example We prove that $\vdash \forall x \forall y(x = y \rightarrow y = x)$:

$$\begin{array}{lll} 1. & x = y & \text{ass.} \\ 2. & x = x & I_= \\ 3. & y = x & 1, 2, E_= \text{ (!)} \\ \hline 4. & x = y \rightarrow y = x & I_\rightarrow \\ 5. & \forall y(x = y \rightarrow y = x) & 4, I_\forall \\ 6. & \forall x \forall y(x = y \rightarrow y = x) & 5, I_\forall \end{array}$$

Notice that the formula introduced at line 2 can be written as $[x/z]z = x$: x has the property to be identical to x. Using $E_=$ one can conclude from this and the assumption that x is identical to y that this property of being identical to x also belongs to y.

Exercise 7 *Prove:*

(a) $\vdash \forall x(x = x)$

(b) $\vdash \forall x \forall y \forall z((x = y \wedge y = z) \rightarrow x = z)$

1.3.1 Leibniz's principle

The introduction and elimination rule for = don't come out of the blue. They can be motivated by Leibniz's Principle: two objects are identical (one and the same object) iff they have all properties in common. The elimination rule for =, $E_=$, is nothing but a formal implimentation of this principe.

For the introduction rule $I_=$ matters are less perspicuous: Leibniz's Principe suggests an introduction rule of the following form: It is allowed to introduce a formula $s = t$ if you have shown that *every property of s is also possessed by t, and vice versa.* But how turn the italicized condition into a formal rule of proof?

This would be wrong: It is allowed to introduce $s = t$ if you have shown for every formula φ in which the variable x occurs free that $[s/x]\varphi \rightarrow [t/x]\varphi$ and that $[t/x]\varphi \rightarrow [s/x]\varphi$. This would be wrong for two reasons. First, this way you get an introduction rule that you can only apply after infinitely many previous steps, because there are infinitely many formulas φ for which you have to show that that $[s/x]\varphi \rightarrow [t/x]\varphi$ and that $[t/x]\varphi \rightarrow [s/x]\varphi$. Moreover, even if you could make these infinitely many steps, it would still not be valid to conclude that $s = t$, because there still could be a property — one that is not expressible in the language — possessed by one but not by the other. These difficulties can be solved by proceeding in the same manner here as in the case of the introduction rule for the universal quantifier $I_\forall$: Let P be an predicate symbol that does not occur in a premise or in an active assumption — P can play the role of an *arbitrary* property. It is allowed to infer that $s = t$ if you have shown that for this "arbitrary" P it holds that $(P(s) \rightarrow P(t)) \wedge (P(t) \rightarrow P(s))$. Schematically:

Alternative introduction rule for the identity sign

$$\begin{array}{lcr} & \vdots & \\ m. & (Ps \rightarrow Pt) \wedge (Pt \rightarrow Ps) & \\ & \vdots & \\ n. & s = t & I'_= \end{array}$$

Requirement:

1. P does not occur in a premise or an assumption that is active at line n.

This *alternative* introduction rule $I'_=$ differs a lot from the official rule $I_=$ introduced above. It will be clear that $I'_=$ implies $I_=$. Conversely, it turns out that $I'_=$ is a (so-called) *derived* rule in the system with $I_=$. More precisely, the following theorem holds.

Proposition 1 *If $\Delta \vdash (Ps \rightarrow Pt) \wedge (Pt \rightarrow Ps)$ for a predicate P that occurs nowhere in the formulas of Δ, then $\Delta \vdash s = t$.*

Exercise 8 *Prove proposition 1.*
(Hint: Consider a derivation of $(Ps \rightarrow Pt) \wedge (Pt \rightarrow Ps)$ from Δ. In this derivation, replace every (sub)formula of the form Pt', for t' any term, by the formula $s = t'$.[6] It is easy to check that by so doing one obtains a new derivation, this time of the formula $s = s \leftrightarrow s = t$. Now, ...)

[6]Consider first the case in which there are no variables in s. What can go wrong if there are variables in s, and how can this problem be solved?