# 185.A09 Advanced Mathematical Logic

www.volny.cz/behounek/logic/teaching/mathlog13

Libor Běhounek, behounek@cs.cas.cz

Lecture #1, October 15, 2013

## Organizational matters

Study materials will be posted and course information regularly updated at the course webpage, www.volny.cz/behounek/logic/teaching/mathlog13

Further reading (not required—just for interested students):

- E. Mendelson: *Introduction to Mathematical Logic* (4th ed.), Chapman & Hall 1997.
- J.R. Shoenfield: *Mathematical Logic* (2nd ed.), A K Peters/CRC Press, 2001.
- J. Barwise (ed.): *Handbook of Mathematical Logic.* North-Holland, 1977.

## Mathematical logic

*Mathematical logic* = the study of mathematized (formal, symbolic) logic and mathematical properties of formal logical and mathematical systems (*metamathematics*).

In mathematical logic, logical notions are modeled by mathematical structures and their properties or relationships; problems in mathematical logic regard these mathematical structures, not the adequateness of such modeling, which is the domain of *philosophical logic.*

Since the 1940–50's, the following disciplines have been established as the main fields of mathematical logic: (i) model theory, (ii) proof theory, (iii) recursion theory, and (iv) set theory.

Traditionally, mathematical logic only deals with: (i) classical logic, (ii) intuitionistic logic, and (iii) the logics of strength intermediary between the two (called *superintuitionistic* or *intermediary logics*), mainly in their first-order variants. However, the notions and methods of mathematical logic have been successfully applied to a wide range of other logics (esp. higher-order, modal, substructural, and fuzzy).

This course will mostly focus on classical model theory and recursion theory: the aim is to cover important concepts of classical metamathematics, esp. topics related to Gödel theorems and the (un)decidability of logical systems.

## Object language and metalanguage

When speaking in one language about another language, we call the former *metalanguage* and the latter *object language.*

**Example.** When explaining German grammar in English, German is the object language and English is the metalanguage.

NB: What is the metalanguage and the object language is context-dependent.

The metalanguage can be the same language as the object language (e.g., when explaining English grammar in English).

When speaking about a metalanguage, we use a *meta-meta-language.* The hierarchy of metalanguages can be iterated through all ordinal numbers. (In this course, however, we shall mostly use just one or two levels of metalanguage.)

The distinction between the object language and metalanguage is important for avoiding paradoxes based on conflating the two, e.g., the *Berry paradox:* "The first natural number not definable in less than twelve words" is defined in eleven words.

In mathematical logic, the object language is usually a *formal* language in which some formal logical system is formulated. The metalanguage can be ordinary natural language, but quite often even the metalanguage is formal (usually, the language of classical first-order logic). Also quite often in mathematical logic, the object language and the metalanguage coincide (most usually, both are the language of classical first-order logic).

In writing, the distinction between the language and metalanguage can be made explicit, for example, by quotation marks, by using different graphical symbols in each (e.g., denoting implication by $\to$ in the object language and by $\Rightarrow$ in the metalanguage), by using different typeface in each (e.g., italics for object-language symbols or boldface for metalanguage symbols). Often, however, no graphical distinction is made in texts on mathematical logic and the correct interpretation is left to the competent reader.

The distinction between the object language and metalanguage transfers also to concepts formulated in the languages: e.g., the *object theory* vs. *metatheory.* In typical situations in mathematical logic we thus generally distinguish the *object level* and the *metalevel* of concepts. The metalevel study of mathematics (or formal logic) by mathematical methods is called *metamathematics.*

## Syntax and semantics

In general, *syntax* is the study of expressions of a given language and the relationships between them regardless of their meaning, while *semantics* studies the meaning of the expressions, i.e., the relationships between the expressions and the objects they denote.

In logic, syntax regards the symbols and expressions (e.g., formulae, sequents, etc.) of a given formal language (usually, the language of some logical system), while semantics deals with the interpretation of these expressions (usually in some mathematical structure).

In mathematical logic, syntax is mostly studied in proof theory, while semantics in model theory.

In mathematical logic, both syntax and semantics are mathematized:

- Syntactic expressions of formal languages are regarded as sequences (or rooted trees or other structures) of formal symbols (the formal symbols themselves are also encoded and represented by mathematical objects—e.g., sets or numbers)
- The denotations of syntactic expressions are defined to be elements of some mathematical structure (e.g., an algebra, a relational system, etc.)

- The relation of denotation is regarded as a function from the set of syntactic expressions to the mathematical structure.

Being mathematical objects, the syntactic expressions and semantic denotations are available for formal mathematical methods. Various syntactic and semantic notions then become definable as mathematical properties, relations, structures, or other objects.

Like in the language–metalanguage distinction, corresponding symbols in syntax and semantics can be graphically distinguished (e.g., by using $\rightarrow$ for the symbol of implication in syntax and $\Rightarrow$ for the function interpreting implication in semantics, or $\bar{0}$ for the syntactic constant 'zero' in formulae and 0 for the object representing zero in models of natural numbers). Often, though, no graphical distinction is made and the correct interpretation is left to the competent reader.

## The syntax of classical propositional and first-order logic

The basic knowledge of classical propositional and first-order logic is assumed in this course (see any standard textbook in logic for definitions). In order to fix notation and to introduce some conventions, we repeat some definitions here.

Propositional variables will be denoted by $p, q, \ldots$, formulae by $\varphi, \psi, \ldots$.

Our only primitive propositional connectives will be implication $\rightarrow$ and negation $\neg$. Other basic propositional connectives will be defined as follows:

$$
\begin{aligned}
\varphi \vee \psi &\quad \equiv_{\mathrm{df}} \quad \neg\varphi \rightarrow \psi \\
\varphi \wedge \psi &\quad \equiv_{\mathrm{df}} \quad \neg(\varphi \rightarrow \neg\psi) \\
\varphi \leftrightarrow \psi &\quad \equiv_{\mathrm{df}} \quad (\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi)
\end{aligned}
$$

In formulae, $\leftrightarrow$ and $\rightarrow$ will take the lowest and $\neg$ the highest precedence; i.e., $\neg p \wedge q \rightarrow r$ will be understood as $((\neg p) \wedge q) \rightarrow r$.

We shall adopt the following axiomatic system for classical propositional logic:

$$
\begin{aligned}
&\varphi \rightarrow (\psi \rightarrow \varphi) \\
&(\varphi \rightarrow (\psi \rightarrow \chi)) \rightarrow ((\varphi \rightarrow \psi) \rightarrow (\varphi \rightarrow \chi)) \\
&(\neg\varphi \rightarrow \neg\psi) \rightarrow (\psi \rightarrow \varphi)
\end{aligned}
$$

with the rule of modus ponens (from $\varphi$ and $\varphi \rightarrow \psi$ infer $\psi$), where $\varphi, \psi, \chi$ are any propositional formulae.

Note: Many metamathematical definitions and proofs (e.g., those by induction on the complexity of formulae or the length of proof) need to take cases on all propositional connectives or all axioms and rules. Therefore it is advantageous to keep the number of primitive connectives, axioms, and inference rules small. Our choice of connectives and axiomatic system makes a compromise between minimality and readability (classical propositional logic can also be axiomatized with a single connective, single axiom, and a single inference rule).

This axiomatic system has the form of a *Hilbert-style* calculus (i.e., has formulae as basic derivable expressions and a small number of derivation rules). There are other kinds of formal calculi for classical logic—e.g., natural deduction calculi or (Gentzen-style) sequent calculi. While finding proofs *in* Hilbert-style calculi is difficult (cf. the example below), metamathematical proofs *on* Hilbert-style calculi are usually shorter and easier because of their simplicity.

As usual, a *proof* (in the above calculus) of a formula $\varphi$ from a set of formulae $\Gamma$ is a finite sequence of formulae, all of which are either (i) axioms of the calculus, (ii) elements of $\Gamma$, or (iii) are derived from the previous formulae by a derivation rule (i.e., by modus ponens, which is our only derivation rule), and whose last element is $\varphi$. If there is a proof of $\varphi$ from the empty set of premises, we say that $\varphi$ is *provable* in (or a *theorem* of) the calculus.

**Example.** The following sequence of formulae is a proof of the formula $p \to p$ in the above calculus:

| | | |
|---|---|---|
| 1. | $(p \to ((p \to p) \to p)) \to ((p \to (p \to p)) \to (p \to p))$ | (an instance of the second axiom) |
| 2. | $(p \to ((p \to p) \to p))$ | (an instance of the first axiom) |
| 3. | $(p \to (p \to p)) \to (p \to p)$ | (modus ponens applied to 1, 2) |
| 4. | $p \to (p \to p)$ | (an instance of the first axiom) |
| 5. | $p \to p$ | (modus ponens applied to 3, 4) |

The syntax of classical first-order predicate logic consists of:

- Propositional connectives as above
- The quantifier $\forall$ ($\exists$ is defined by $(\exists x)\varphi \equiv_{\mathrm{df}} \neg(\forall x)\neg\varphi$)
- An infinite countable set of *individual* (or *object*) *variables* $x, y, \ldots$
- Function symbols $F, G, \ldots$, each assigned an arity $\mathrm{ar}(F) \geq 0$; nullary function symbols are also called *individual* (or *object*) *constants*
- Predicate symbols $P, Q, \ldots$, each assigned an arity $\mathrm{ar}(P) \geq 0$; nullary predicate symbols are also called *propositional constants*

A *predicate language* is a set of predicate and function symbols.

Predicate logic can be defined with or without a distinguished binary predicate $=$, regarded as a logical symbol (i.e., not part of the predicate language, which collects *extralogical* symbols subject to interpretation in semantic structures; the semantics of $=$, on the other hand, is fixed as the identity of elements; however, $=$ is often listed alongside the extralogical symbols of the language). In predicate logic without equality, predicate languages are assumed to be non-empty. In predicate logic with equality, $=$ ensures the existence of formulae even if the predicate language is empty.

**Example.** The following formal languages are often encountered in mathematics and studied in metamathematics.

- The *language of set theory* consists of the single binary predicate $\in$. (Other set-theoretical predicate and function symbols can be regarded as definable in terms of $\in$; for the metamathematical proofs it is expedient to keep the primitive language small.)
- The *language of elementary group theory* consists of the binary predicate $=$, the binary function $\cdot$ (multiplication), the unary function $^{-1}$ (inverse), and the individual constant 1. Similarly for the language of elementary theory of fields, lattices, etc.
- The *language of arithmetic* consists of the individual constant 0, the unary function $s$ (of successor), the binary functions $+$ and $\cdot$, and the binary predicates $=$ and $\leq$.

The set of *terms* of a predicate language $L$ (or *$L$-terms*) is the smallest set such that:

- Every individual variable is an $L$-term
- If $t_1, \ldots, t_n$ are $L$-terms and $F$ is an $n$-ary function symbol in $L$, then $F(t_1, \ldots, t_n)$ is an $L$-term

*Atomic $L$-formulae* are the expressions of the form $P(t_1, \ldots, t_n)$, where $t_1, \ldots, t_n$ are $L$-terms and $P$ is an $n$-ary predicate symbol in $L$. The set of all *$L$-formulae* is the smallest set such that:

- Every atomic $L$-formula is an $L$-formula
- If $\varphi$ and $\psi$ are $L$-formulae and $x$ an individual variable, then $\varphi \to \psi$, $\neg\varphi$, and $(\forall x)\varphi$ are $L$-formulae

If the predicate language is fixed or known from the context, we simply speak of terms and formulae.

Note: Since the notion of formula is defined by recursion, metamathematical proofs and definitions involving formulae are often done by induction on the recursive complexity of a formula.

*Free* and *bound* occurrences of individual variables in formulae are defined as follows:

- Every occurrence of an individual variable in an atomic formula is free
- If an occurrence of $x$ in $\varphi$ is free (bound, resp.), then the corresponding occurrence of $x$ in $\varphi \to \psi$, $\psi \to \varphi$, and $\neg\varphi$ is free (bound, resp.) as well
- All occurrences of $x$ in $(\forall x)\varphi$ are bound
- If $y$ is a variable different from $x$, then an occurrence of $y$ in $(\forall x)\varphi$ is free iff the corresponding occurrence of $y$ in $\varphi$ is free; otherwise it is bound

A formula is *open* if it contains no bound variables, and *closed* (or a *sentence*) if it contains no free variables. The universal closure of $\varphi$ is the formula $(\forall x_1)\ldots(\forall x_n)\varphi$, where $\{x_1, \ldots, x_n\}$ is the set of free variables in $\varphi$. A term $t$ is *substitutable* for $x$ in $\varphi$ (or: $t$ is free for $x$ in $\varphi$) if no variable occurring in $t$ becomes bound in $\varphi(t/x)$, where $\varphi(t/x)$ denotes the formula resulting from substituting $t$ for $x$ in $\varphi$.

We shall adopt the following axiomatic system for classical first-order logic. The *logical axioms* are all $L$-formulae instantiating the axioms of classical propositional logic, plus:

- $(\forall x)\varphi \to \varphi(t/x)$, for $t$ free for $x$ in $\varphi$ ("specification")
- $(\forall x)(\varphi \to \psi) \to (\varphi \to (\forall x)\psi)$, if $x$ is not free in $\varphi$ ("distribution")

The inference rules are:

- Modus ponens: from $\varphi$ and $\varphi \to \psi$ infer $\psi$
- Generalization: from $\varphi$ infer $(\forall x)\varphi$

Predicate logic with equality adds the following axioms for all $P, F \in L \cup \{=\}$:

- $x = x$
- $x_1 = y_1 \wedge \ldots \wedge x_n = y_n \to (P(x_1, \ldots, x_n) \leftrightarrow P(y_1, \ldots, y_n))$
- $x_1 = y_1 \wedge \ldots \wedge x_n = y_n \to (F(x_1, \ldots, x_n) = F(y_1, \ldots, y_n))$

A *theory* is a set of formulae in a given language $L$. The elements of $T$ are called the *axioms of* $T$.

**Example.** The usual axioms for groups, set theory ZFC, Peano arithmetic, etc., are formal theories in this sense.

A *proof* of $\varphi$ in a theory $T$ is a sequence of formulae ending with $\varphi$, each member of which is a logical axiom, an axiom of $T$, or is derived from preceding formulae by an inference rule. (A more general definition uses well-founded trees of formulae; due to the compactness theorem, sequences are sufficient for classical first-order logic.)

A formula $\varphi$ is *provable* in a theory $T$ (written $T \vdash \varphi$) if there is a proof of $\varphi$ in $T$. A theory is *deductively closed* if it satisfies $T \vdash \varphi$ iff $\varphi \in T$ for all $\varphi$. A theory $T$ is *inconsistent* if $T \vdash \varphi$ for all $\varphi$; otherwise it is *consistent.* If $T \vdash \varphi \leftrightarrow \psi$, then we say that $\varphi$ and $\psi$ are equivalent in $T$. A theory $T$ is *syntactically complete* if $T \vdash \varphi$ or $T \vdash \neg\varphi$ for each closed formula $\varphi$.

**Example.** Elementary group theory is syntactically incomplete, as, for instance, the commutativity law is neither provable nor refutable in the theory. (To prove this, we'll need the Soundness Theorem and the observation that not every group is Abelian.)

A theory $S$ *extends* (or *contains*) $T$ if the language of $S$ contains the language of $T$ and $S$ proves all axioms of $T$. An extension $S$ of $T$ is *conservative* if $S \vdash \varphi$ implies $T \vdash \varphi$ for all $\varphi$ in the language of $T$.

**Exercise.** *Prove in the above axiomatic system for classical first-order logic with equality that equality is symmetric and transitive, i.e., give formal proofs of the formulae* $x = y \to y = x$ *and* $x = y \land y = z \to x = z$.