Mathematical Logic

Lecture 1

Dr Marco Benini

marco.benini@uninsubria.it

Dipartimento di Scienza e Alta Tecnologia
Università degli Studi dell'Insubria

a.a. 2016/17

Syllabus

Bureaucracy:

- Introduction
- Program
- Examination
- Timing
- Questions

A brief introduction to logic:

- What does it speak about?
- Syntax, intended meaning, semantics
- An infinite variety of logics
- Foundational issues
- Soundness and completeness

Introduction

Mathematical logic is a field of Mathematics which studies the deduction process, and the foundations of the whole discipline.

This course will introduce mathematical logic from the very beginning, assuming a minimal knowledge of elementary mathematics.

Also, the material of the course is, more or less, standard, and most introductory textbooks will cover it. For the purposes of this course, slides and lecture notes will be made available to students after every lesson.

The course is in English.

Program

The course takes 64 hours, and its content will be an introduction to classical logic, with a glimpse to other logical systems.

The detailed program is

- *Propositional logic*: language, deduction system, semantics, soundness, completeness;
- *First-order logic*: syntax, semantics, soundness, completeness;
- *Set theory*: fundamental axioms, ordinals, cardinals, transfinite induction, axiom of choice, continuum hypothesis;
- *Constructive mathematics*: intuitionistic logic, computable functions, λ calculi, propositions as types;
- *Limiting results*: Peano arithmetic, Gödel's incompleteness theorems, natural incompleteness results.

Texts

Since this is the second year this course is taught, there is no textbook available in advance. A draft of the textbook is available on the course website and, from time to time, it will grow as the lessons will proceed.

All the slides, along with the lecture notes will become available roughly after each lesson at the course website:

`http://marcobenini.wordpress.com/lectures/mathematical-logic`

Also, at the end of each lesson, references to articles, texts, and other resources which may be of interest to those interested in learning more, will be available. While the content of slides is *mandatory*, looking at references is *optional*. Also, the lecture notes will provide the same material as the slides, eventually complemented with exercises: while it is not mandatory to study on the lecture notes, they could be a big aid.

Examination

The examination will be oral. It will require to perform simple exercises, like proving a theorem using a formal deductive system, and to state, discuss, and prove the results explained during the course.

The examination will be, at the student's choice, either in Italian or in English.

Informally, a student may take the examination by fixing an appointment: this can be done at every time, after the end of the course. Formally, examinations can be registered only during the dates scheduled in the official calendar: students **must** subscribe the date to be able to register their marks. Students are strongly encouraged to plan when to take examinations, and to fix an appointment in advance. Then, they can register the result whenever they prefer, within 18 months from the beginning of the course.

As usual, independently from the results, repeating an examination cancels the previous ones.

Examination

Although it is not mandatory, there will be four intermediate assignments during the course.
They will take place during the lesson time, and they will cover

1. propositional logic
2. first-order logic
3. set theory and constructive mathematics
4. limiting results

Students willing to take them, can avoid the examination: each assignment will get a mark, and the average will be the final mark. Rules for registration are the same as for regular examinations.

Timing

The schedule of lessons is fixed, and it cannot be easily changed. In general, a lesson will start 10 minutes after the official time, and it will finish 15 minutes before the official time, so that students can move between classrooms.

There will be one 15 minutes pause during the lessons.

Questions

Questions are welcome. Please, do not hesitate to ask questions when you do not understand something during a lesson.

Questions could be asked also before the start of a lesson, or after the end.

Another possibility is the ask questions by email: in case write at the address

`marco.benini@uninsubria.it`

specifying your name, the course, and the question. If possible, try to use your *official* email from `uninsubria`.

There are no office hours in this course: students have to fix an appointment. Please, do so only if you really think there is no other way to solve your problem: although I am usually available to receive students during the course time, when I am not teaching, it is often the case that I am not in Italy, so, please, use this as your last resource.

Your teacher

I am a researcher in Mathematical Logic. This means that my main job is to think, and, sometimes, to prove novel results in this field of Mathematics.

Teaching is part of my academic duties, but is not my first occupation.

As a logician, my interests lie in the interplay between truth and computability. In fact, I investigate mainly constructive logical systems, which have nice computational properties, and my favourite playground, the 'universe' I work within, is topos theory, a branch of category theory.

For more, please visit my web page:

`http://marcobenini.wordpress.com`

Mathematical Logic

Mathematical logic

Mathematical logic studies the mathematical deduction process and the notion of truth, at large.

Logic is an ancient part of Mathematics: its origins go back to Aristotle, while its mathematical foundations can be traced in the work of Boole, Frege, Cantor, Russell, Hilbert, Gödel, ...

Since Gödel Incompleteness result, the discipline underwent a huge development, and, today, it is a very active part of contemporary Mathematics, with application in Computer Science and Philosophy.

Since this is a first course in mathematical logic, we will stop after proving the incompleteness results. Here and there, hints about future developments will be given, but the course sticks on the classical track.

Logic is formal

Consider arithmetic as a guiding example. When expressing this theory in logical terms, you will have three main levels to look at it:

- syntax
- semantics
- intended interpretation

Logic keeps the intended interpretation in the background, and it focuses on the study of syntax and semantics.

Also, the syntax and the semantics are *formal*: although this could be boring, and, in some cases, a burden to get to results, it is also the fundamental tool of logic. If you don't like it, well, you are in the wrong place!

Syntax

The *syntax* is the way we write down things.

For example, $1+2$ is an *expression*, and $1+2=5$ is a formula. The *language* of a theory, e.g., arithmetic, is the collection of rules allowing to write all the possible expressions and formulae.

Also, since we are interested in proving theorems, which are formulae, eventually depending on other formulae, the hypotheses, we need a way to write proofs. The way to construct proofs is, again, formal, and it is described by a *deductive system*, a collection of axioms and rules.

Together, the language and the deductive system form the *syntax* of a theory. Syntactical reasoning is **the** way to think inside a logical theory, the only one which can be studied.

Intended interpretation

The intended interpretation of a theory is the informal, intuitive way to understand a (logical) theory.

For example, when we say that ‘arithmetic studies the properties of integer numbers’, we should read this sentence as ‘the formal theory of arithmetic, that is, its syntax, has the properties of integers as its intended interpretation’, and we assume to know what does it mean to be a property, and what is the shape of integers.

In mathematical logic, we keep the intended interpretation in the background: we are interested in a syntax which allows to express what we intend, e.g., by ‘property’ or by ‘integer’, and we are interested in a formal way to say when a formula is true, which should correspond to a property being valid in the intended interpretation.

Semantics

The semantics is the formal way to attribute a meaning to a given syntax.

We are interested in semantic systems which, in some sense, capture the intended meaning of our theories. For example, in arithmetic, we would like a semantics saying that $1+2=3$ is a true formula, while $1+2=5$ is a false formula.

Usually, a semantics, defines a universe, where expressions are interpreted, and a notion of truth and falsity, which are used to distinguish between valid and invalid properties.

We will see many examples of semantics, in this course, and we will see that a *good* agreement between the syntax and the semantics is what we will constantly search for.

An infinite variety of logics

Surprisingly, at first, there is not just one logic.

The fundamental connectives of logic are $\wedge$ (and), $\vee$ (or), $\supset$ (if. . . then. . .), $\neg$ (not), $\top$ (true), and $\bot$ (false). The fundamental quantifiers are $\forall$ (for all), and $\exists$ (exists).

But there are interesting logics allowing for other connectives and quantifiers: for example, *modal* logics have the connectives $\Box$ (necessity) and $\Diamond$ (possibility).

Temporal logics have the connectives $\Box$ (true from now on) and $\Diamond$ (will become true).

We will not study logics using other connectives than the fundamental ones in this course.

An infinite variety of logics

A logical system may deal with expressions, like arithmetic, or it may speak only of formulae. In the latter case, the system is said to be *propositional*.

On the contrary, we may imagine a system that speaks of elements of some universe. In particular, when we allow to quantify only over elements, we will say that the system is *first-order*.: $\forall x.\mathsf{even}(x) \vee \mathsf{odd}(x)$ is a formula of arithmetic.

On the contrary, when we allow to quantify over collections of elements, we will speak of *higher-order* systems. For example, the formula $\forall S.\exists n.\max S < n \wedge \min S > -n \supset 0 \in S$ is a second-order formula since we quantify over S, which stands for a set of integers.

In this course, we will study propositional and first-order systems.

A rule of thumb says that all the mathematics developed before the 20^{th} century could be expressed as a collection of first-order theories.

An infinite variety of logics

When we want to reason about a mathematical system, we may want to have some aspects in the 'structure' of the reasoning system: if we prefer not to have time as an explicit parameter, we can move it inside the logic.

But which time? Discrete or continuous? A linear order or a branching tree of possible worlds? And this is just the beginning... in fact, all of these 'times' have real applications in Physics or Computer Science.

But even with the standard connectives we have choices to make. Consider negation: what means to prove $\neg A$? Are we satisfied by saying that A is false? Or do we pretend that a counterexample to A exists? Or, even more, do we pretend that a counterexample to A has to be inside the proof of $\neg A$?

Classical logics

The standard connectives and quantifiers must be coupled with axioms and rules so to deduce formulae from other formulae.

For example, $\forall x.x = x$ is an axiom stating that equality is reflexive. And

$$\frac{A \quad B}{A \wedge B}$$

is a rule saying that, from the formulae A and B, we can deduce $A \wedge B$.

A logic is said to be *classical* when it allows to deduce $A \vee \neg A$ for any formula A. This principle is called *tertium non datur* or, also, the *Law of Excluded Middle*.

This principle has a number of consequences, for example, in arithmetic it allows to define functions which are not computable. So, adopting it is a choice, and there are systems which do not.

In this course, we will limit our study to classical systems, with one big exception: intuitionistic logic, which is, in some sense, the logic of computable functions.

Foundational issues

One of the big motivations for studying mathematical logic lies in the foundational problem: is Mathematics coherent?

In fact, as we will see in the end of this course, there is no hope to answer such a question within mathematics. But, still, relative coherence is an important question and it can be answered: is it impossible to deduce a statement and it negation in a given logical system, assuming that another theory is coherent?

As we will see, this question can be addressed, and some of its consequences are surprising: these will be presented at the end of this course.

Foundational issues

As a matter of fact, most branches of Mathematics could be developed using set theory plus classical logic as a framework: for example, arithmetic can be derived by identifying natural numbers with some special sets, and arithmetical operations become specific functions.

Since we have not to add any axiom or rule, but just definitions, that is, we add names, shorthands if you prefer, to the language, we could say that set theory is expressive enough to model arithmetic.

The pursue for a universal theory, one allowing to model every mathematical theory, is impossible to achieve, as we will prove in this course, but, still, some theories, like set theory, are close enough to allow us to reason on almost the whole Mathematics. In this course, we will discuss set theory to some extent, although we will not study any other such 'universal' theory.

Soundness and completeness

The first and fundamental intent of a logical system is to derive the true sentences. To this aim, a deductive system is provided by the syntax, and a notion of truth is provided by the semantics.

It is worth noticing that different semantics may provide different notions of truth, and, in fact, truth is not universal in logic: it strictly depends on the semantics we will adopt. And yes, the same theory may have different semantics, not necessarily compatible.

This raises two major questions:

- is it the case that every formula we may prove is true?
- is it the case that every formula which is true, admits a proof in the deductive system?

Soundness and completeness

The first property is called *soundness*: we are not interested in non-sound deductive systems. A fundamental requirement for a syntax is to forbid deriving false consequences from true hypotheses. But we must prove that a syntax is sound with respect to a given semantics.

The second property is *completeness*: a syntax is a perfect description of a semantics when it allows to prove every true statement and to show that every false statement has a counterexample. We will see that completeness, as stated, is a very strong property. More, we will show that the majority of naturally interesting theories cannot be complete in the above sense, a shocking fact that changed the history of Mathematics.

There are many other properties of interest in logic, and, from time to time we will mention them, as appropriate. But soundness and completeness are the most fundamental ones, and we will focus on them in this course.

References

For those interested in the history of logic, and its relations to Mathematics, a nice, short book is *Piergiorgio Odifreddi*, La matematica del Novecento—Dagli insiemi alla complessità, Piccola Biblioteca Einaudi, Einaudi, (2000), ISBN 88-06-15153-3.

There are many introductory textbooks of mathematical logic and a few important reference books. I would like to mention the comprehensive guide, *Jon Barwise*, Handbook of Mathematical Logic, Studies in Logic and the Foundations of Mathematics 90, North-Holland, (1977), ISBN 0-444-863888-5.

I do not have a preferred textbook, but I suggest the following notes by Prof. Helmut Schwichtenberg:

`http://www.mathematik.uni-muenchen.de/~schwicht/lectures/logic/ws03/ml.pdf`

Mathematical Logic

Lecture 2

Dr Marco Benini

marco.benini@uninsubria.it

Dipartimento di Scienza e Alta Tecnologia
Università degli Studi dell'Insubria

a.a. 2016/17

Syllabus

Propositional logic:

- Language
- Induction
- Intended interpretation
- Deduction system
- Informal meaning
- Examples

Propositional logic

In this lesson, we want to introduce classical propositional logic.

We will start from its syntax, and its intended meaning.

The idea is that a proposition stands for a *truth value*, either *true* or *false*. Composite propositions will derive their truth value from their components, while basic propositions will have a truth value which depends on the world where they are interpreted in.

For example, the sentence 'Socrates is a man' may be true or false, as Socrates may be the ancient Greek philosopher, or a cat. On the other side, 'If Socrates is a man then Socrates is a mortal' is true when Socrates is both a man and mortal, but also when Socrates is not a man, and it is false when Socrates is an immortal man.

Language

Definition 2.1 (Formula)

Let $\mathcal{V}$ be an infinite set of symbols, called *variables*, not containing '(', ')', '$\top$', '$\bot$', '$\wedge$', '$\vee$', '$\supset$', '$\neg$'.
Then, a *formula* is inductively defined as

1. a variable $x \in \mathcal{V}$ is a formula;
2. $\top$, spelt *true*, and $\bot$, *false*, are formulae;
3. if A is a formula, so is $(\neg A)$, *not*, *negation*;
4. if A and B are formulae, so are $(A \wedge B)$, *and*, *conjunction*, $(A \vee B)$, *or*, *disjunction*, and $(A \supset B)$, *implication*.

Notice how A and B above are not part of the language, but are variables in the metalanguage—we will be mostly informal about the metalanguage, i.e., the language we use to describe the logical language.

Language

To simplify the notation, we use a number of abbreviations:

- outermost parentheses are not written: $x \wedge y$ instead of $(x \wedge y)$;
- conjunction and disjunction have a higher precedence over implication: $x \wedge y \supset z \vee w$ instead of $((x \wedge y) \supset (z \vee w))$;
- negation has a higher precedence over conjunction, disjunction, and implication: $\neg x \wedge \neg y$ instead of $((\neg x) \wedge (\neg y))$;
- lowercase letters, when not specified otherwise, stand for variables.
- uppercase letters, when not specified otherwise, stand for objects in the metalanguage.

An important point to remark is that the definition of formula is by induction. So, we can use this structure to define new notions or to prove properties of formulae.

On induction

Induction is a powerful tool. Induction allows to define new concepts and to prove statements about a collection of elements.

Informally, to show that a property $P(x)$ holds for every possible value of x, one could substitute x with any possible value. This is, generally, impractical, and impossible when the domain is infinite. But there are many cases when, although the domain is infinite, it can be generated by a finite number of rules. For example, there are infinite propositional formulae, but they are generated according to a finite grammar, the one of Definition 2.1.

In those cases, instead of proving a property for any value, we can show that the property holds for any case of the (inductive) definition. This amounts to show that, whatever value is generated by the rules, it will satisfy the property. Since every value is generated, all possible cases are covered.

Definition works similarly: a concept depends on a value in a domain. Thus, all the possible instances are generated by all the values. Or, inductively, by generating all the instances via the grammar that generates all the values.

Language

As an example of inductive definition, let's define the notion of *subformula*:

Definition 2.2 (Subformula)

Given a formula A on the set V of variables, B is a *subformula* of A if and only if B belongs to the set $S(A)$ inductively defined as

1. if $A \in V$, $A \equiv \top$, or $A \equiv \bot$, then $S(A) = \{A\}$;
2. if $A \equiv B \wedge C$, $A \equiv B \vee C$, or $A \equiv B \supset C$, then $S(A) = \{A\} \cup S(B) \cup S(C)$;
3. if $A \equiv \neg B$, then $S(A) = \{A\} \cup S(B)$.

We may equivalently say that B *occurs* in A, meaning that B is a subformula of A.

In general, the symbol $\equiv$ in the meta-language means 'literally equal', i.e., written in exactly the same way.

Intended interpretation

Informally, a *truth value* is either true or false.

- A variable stands for some truth value.
- $\top$ denotes true.
- $\bot$ denotes false.
- $A \wedge B$ is true when both A and B are true, and false otherwise.
- $A \vee B$ is true when A is true, or B is true, or both are true, and false when both A and B are false.
- $A \supset B$ is true if, when A is true, so is B, and it is true also when A is false. It is false when A is true but B is false.
- $\neg A$ is true exactly when A is false.

In general, the truth value of a formula depends on the values of its variables. Sometimes, it happens that a formula is true independently from the value of its variables, e.g., $x \supset x$ is true whatever truth value x may assume.

Logic is mainly concerned in the study of those formulae which are true independently from the values of their variables.

Natural deduction

An obvious way to discover whether a formula is true, is to try all the possible values for the variables occurring in it.

But there are three main drawbacks in this strategy:

- the strategy is exponential: if there are n distinct variables in a formula, we have to try 2^n possible assignments.
- the strategy does not scale to other logical systems. For example, take arithmetic: it is unfeasible to show the truth of a formula trying all the possible values for its variables, as each of them stands for a natural!
- the strategy does not provide any insight: we have no idea why the formula holds, except that it exhaustively satisfies all the possible assignments. In particular, we do not know which axioms in our theory are required so to make the property true.

What we want is a notion of *proof*: a way to reason that, starting from some basic accepted facts, and adopting a series of accepted rules, allows us to conclude that the formula is true.

Natural deduction

Definition 2.3 (Theory)

Fixed a language, a *theory* T is a set of formulae, each one usually referred to as an *axiom*.
When $T = \emptyset$, we will speak of the theory as *pure logic*.

Definition 2.4 (Proof)

Fixed a language and a theory T in it, a *proof* or *deduction* of the formula A, the *conclusion*, from a set Γ of formulae, the *hypotheses* or *assumptions*, is inductively defined by a set of inference rules summarised in the next slides.
A formula A which is the conclusion of a proof with no assumptions, is called a *theorem* in the theory T.

Natural deduction

The inference rules governing conjunctions are:

$$\frac{A \wedge B}{A}\,\wedge\mathsf{E}_1 \qquad \frac{A \wedge B}{B}\,\wedge\mathsf{E}_2 \qquad \frac{A \quad B}{A \wedge B}\,\wedge\mathsf{I}$$

we have two elimination rules, and an introduction rule.

Those governing disjunctions are:

$$\frac{A}{A \vee B}\,\vee\mathsf{I}_1 \qquad \frac{B}{A \vee B}\,\vee\mathsf{I}_2 \qquad \frac{A \vee B \quad \begin{array}{c}[A]\\ \vdots\\ C\end{array} \quad \begin{array}{c}[B]\\ \vdots\\ C\end{array}}{C}\,\vee\mathsf{E}$$

Natural deduction

Implication and negation are subject to the following rules:

$$\frac{\begin{array}{c}[A]\\ \vdots \\ B\end{array}}{A \supset B}\supset\mathrm{I} \qquad \frac{A \supset B \quad A}{B}\supset\mathrm{E}$$

$$\frac{\begin{array}{c}[A]\\ \vdots \\ \bot\end{array}}{\neg A}\neg\mathrm{I} \qquad \frac{\neg A \quad A}{\bot}\neg\mathrm{E}$$

They are very similar, since, as we will see in the next lesson, negation can be defined from implication.

Natural deduction

True and false are governed by the following rules:

$$\frac{}{\top}\,\top\mathsf{I} \qquad \frac{\bot}{A}\,\bot\mathsf{E}$$

If A is an axiom of the theory T, i.e., if $A \in T$, we are allowed to deduce it:

$$\frac{}{A}\,\mathsf{ax}$$

If A is an assumption, i.e., if $A \in \Gamma$, we can deduce it

$$A$$

Natural deduction

Finally, for every formula A, either A is true or it is false. This is expressed by the Law of Excluded Middle:

$$\frac{}{A \vee \neg A}\ \text{lem}$$

As we will say later in the course, the Law of Excluded Middle is *delicate*, and it has a special status.

In general, whenever possible, we will try to avoid its use in a proof.

Natural deduction

A couple of comments:

- except for the Law of Excluded Middle, the rules come in pairs: any connective is associated to one or more introduction rule, and one or more elimination rule.
- assumptions may be *free* of *discharged*. Free assumptions are real, in the sense that the proof depends on them; discharged assumptions are used to get rid of a local assumption, which does not affect the whole proof. This is best understood looking at the 'implication introduction' rule: to prove $A \supset B$, we locally assume A, and we try to prove B, but the final result does not depend anymore from A.
- discharging is optional: we must not discharge an assumption when a rule does not allow, but we may (or we may not) discharge an assumption if the rules allows to.

When we do not want to specify the proof, we write $\pi : \Gamma \vdash_T A$, meaning that π is a proof of A from the assumptions Γ in the theory T. When the proof is not relevant, we omit the π; when the theory is understood or empty, we omit the T; when the set of assumptions is empty, we omit the Γ.

Natural deduction

Example 2.5

The formula $(p \supset q) \wedge p \supset q$ is a theorem in the pure logic, i.e., in the empty theory. In fact, this is a proof:

$$\dfrac{\dfrac{\dfrac{[(p \supset q) \wedge p]^*}{p \supset q}\wedge E_1 \quad \dfrac{[(p \supset q) \wedge p]^*}{p}\wedge E_2}{q}\supset E}{(p \supset q) \wedge p \supset q}\supset I^*$$

Discharged assumptions are written in square brackets and the superscripts indicate which inference rule discharges them.
In order to say that such a formula is always true, we could write $\vdash (p \supset q) \wedge p \supset q$.

Natural deduction

Example 2.6

The *double negation* law says that p is equivalent to $\neg\neg p$:

$$\dfrac{\dfrac{\dfrac{[\neg p]^{*} \quad [p]^{\dagger}}{\bot}\,\neg\mathsf{E}}{\neg\neg p}\,\neg\mathsf{I}^{*}}{p \supset \neg\neg p}\,\supset\mathsf{I}^{\dagger}$$

$$\dfrac{\dfrac{\dfrac{}{p \vee \neg p}\,\mathsf{lem} \qquad \dfrac{\dfrac{[\neg\neg p]^{\S} \quad [\neg p]^{\ddagger}}{\bot}\,\neg\mathsf{E}}{p}\,\bot\mathsf{E}}{p}\,\vee\mathsf{E}^{\ddagger}}{\neg\neg p \supset p}\,\supset\mathsf{I}^{\S}$$

In general, we say that two formulae A and B are *equivalent* when we can deduce one from the other, or, which is the same, when $A \supset B$ and $B \supset A$.

Summary

$$\frac{A \quad B}{A \wedge B}\,\wedge\mathrm{I} \qquad \frac{A \wedge B}{A}\,\wedge\mathrm{E}_1 \qquad \frac{A \wedge B}{B}\,\wedge\mathrm{E}_2 \qquad \frac{\bot}{A}\,\bot\mathrm{E}$$

$$\frac{A}{A \vee B}\,\vee\mathrm{I}_1 \qquad \frac{B}{A \vee B}\,\vee\mathrm{I}_2 \qquad \frac{A \vee B \quad \begin{array}{c}[A]\\ \vdots\\ C\end{array} \quad \begin{array}{c}[B]\\ \vdots\\ C\end{array}}{C}\,\vee\mathrm{E} \qquad \frac{}{\top}\,\top\mathrm{I}$$

$$\frac{\begin{array}{c}[A]\\ \vdots\\ B\end{array}}{A \supset B}\,\supset\mathrm{I} \qquad \frac{A \supset B \quad A}{B}\,\supset\mathrm{E} \qquad \boxed{\frac{}{A \vee \neg A}\,\mathrm{lem}} \qquad \frac{\begin{array}{c}[A]\\ \vdots\\ \bot\end{array}}{\neg A}\,\neg\mathrm{I} \qquad \frac{\neg A \quad A}{\bot}\,\neg\mathrm{E}$$

Summary

This lesson is fundamental. You have to memorise the inference rules of the previous slide and use them at will.

Although the intended meaning seems obvious, be sure to really understand the way we interpret implication.

Take some time to notice the symmetries among the inference rules:

- except for the Law of Excluded Middle, there are introduction and elimination rules for every connective;
- you cannot introduce falsity;
- you cannot eliminate truth;
- implication and negation are similar;
- conjunction and disjunction are similar.

Take your time to study the examples: at some point, you will be supposed to be able to make proofs as the presented ones.

References

Natural deduction has, in its current format, been presented in the classical text *D. Prawitz*, Natural Deduction, Almqvist & Wiksell, Stockholm, (1965). Recently, this text has been reprinted by Dover.

We will use mainly *John Bell* and *Moshé Machover*, A Course in Mathematical Logic, North-Holland, (1977), ISBN 0-7204-28440 in this course as a general reference. Although it is an old book, it is still a classical reference, and it contains a complete, formal development of all the notions.

For a comprehensive and deep treatment of natural deduction, see *A.S. Troelstra* and *H. Schwichtenberg*, Basic Proof Theory, Cambridge Tracts in Theoretical Computer Science 43, Cambridge: Cambridge University Press, (1996). This book extends far over the content of our course.

Mathematical Logic

Lecture 3

Dr Marco Benini

marco.benini@uninsubria.it

Dipartimento di Scienza e Alta Tecnologia
Università degli Studi dell'Insubria

a.a. 2016/17

Syllabus

Propositional logic:

- Examples
- Proving techniques

Examples

To prove a formula, we need to think backwards: so introduction rules eliminate the main connective from a formula.

The first basic technique is to reduce the formula to prove by applying the only introduction rule which could generate it.

Example 3.1

Prove $\vdash A \supset (B \supset A)$

$$\dfrac{\dfrac{[A]^1}{B \supset A}\supset \mathsf{I}}{A \supset (B \supset A)}\supset \mathsf{I}^1$$

Examples

A useful way to prove a formula, is to keep track of the assumptions we generate in the proving process.

In the last example, we started from

$$A \supset (B \supset A)$$

We tried to simplify the goal to prove by implication introduction rule

$$\frac{B \supset A}{A \supset (B \supset A)} \supset \mathrm{I}$$

and, in the meanwhile, our set of assumptions, which was initially empty, has become $\{A\}$.

We tried to simplify the current goal, obtaining

$$\frac{\dfrac{A}{B \supset A} \supset \mathrm{I}}{A \supset (B \supset A)} \supset \mathrm{I}$$

and, in the meanwhile, our set of assumptions has become $\{A, B\}$.

Examples

And, now, we see that the current goal is in the set of available assumption, so we can close the proof by discharging.

$$\dfrac{\dfrac{[A]^1}{B \supset A}\supset \mathsf{I}}{A \supset (B \supset A)}\supset \mathsf{I}^1$$

It is worth noticing that

- we should remember which rule introduced which assumption, so that discharging could be correctly performed;
- we may have unused assumptions, like B in the example.

Examples

When an assumption is a complex formula, it is worth dismounting it by means of an elimination rule.

Example 3.2

Prove $\vdash (A \supset C) \supset ((B \supset C) \supset (A \vee B \supset C))$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[A \vee B]^1 \quad \dfrac{[A]^2 \quad [A \supset C]^3}{C}\supset E \quad \dfrac{[B]^2 \quad [B \supset C]^4}{C}\supset E}{C}\vee E^2}{A \vee B \supset C}\supset I^1}{(B \supset C) \supset (A \vee B \supset C)}\supset I^4}{(A \supset C) \supset ((B \supset C) \supset (A \vee B \supset C))}\supset I^3$$

Notice how assumptions are local to a subproof. Try to redo this exercise and to understand how assumptions are managed.

Examples

Example 3.3

Prove $\vdash A \vee B = B \vee A$.

We notice that the property is auto-dual

$$\dfrac{\dfrac{[A \vee B]^1 \quad \dfrac{[A]^2}{B \vee A} \vee I_2 \quad \dfrac{[B]^2}{B \vee A} \vee I_1}{B \vee A} \vee E^2}{A \vee B \supset B \vee A} \supset I^1$$

Examples

Example 3.4

Prove $\vdash A \wedge B = B \wedge A$

$$\dfrac{\dfrac{\dfrac{\dfrac{[A \wedge B]^1}{B}\wedge E_2 \quad \dfrac{[A \wedge B]^1}{A}\wedge E_1}{B \wedge A}\wedge I}{A \wedge B \supset B \wedge A}\supset I^1$$

Examples

There could be more than one way to prove a result.

Example 3.5

Prove $\vdash A \vee A = A$

$$\dfrac{\dfrac{[A \vee A]^1 \quad [A]^2 \quad [A]^2}{A}\vee E^2}{A \vee A \supset A}\supset I^1 \qquad \dfrac{\dfrac{[A]^1}{A \vee A}\vee I_1}{A \supset A \vee A}\supset I^1 \qquad \dfrac{\dfrac{[A]^1}{A \vee A}\vee I_2}{A \supset A \vee A}\supset I^1$$

Examples

Example 3.6

Prove $\vdash A \wedge A = A$

$$\dfrac{\dfrac{[A \wedge A]^1}{A}\,\wedge E_1}{A \wedge A \supset A}\,\supset I^1 \qquad \dfrac{\dfrac{[A \wedge A]^1}{A}\,\wedge E_2}{A \wedge A \supset A}\,\supset I^1 \qquad \dfrac{\dfrac{[A]^1 \quad [A]^1}{A \wedge A}\,\wedge I}{A \supset A \wedge A}\,\supset I^1$$

Examples

Example 3.7

Prove $\vdash A \vee (A \wedge B) = A$

$$\dfrac{\dfrac{[A \vee (A \wedge B)]^1 \quad [A]^2 \quad \dfrac{[A \wedge B]^2}{A} \wedge E_1}{A} \vee E^2}{A \vee (A \wedge B) \supset A} \supset I^1 \qquad \dfrac{\dfrac{[A]^1}{A \vee (A \wedge B)} \vee I_1}{A \supset A \vee (A \wedge B)} \supset I^1$$

Examples

Example 3.8

Prove $\vdash A \wedge (A \vee B) = A$

$$\dfrac{\dfrac{[A \wedge (A \vee B)]^1}{A}\,\wedge E_1}{A \wedge (A \vee B) \supset A}\,\supset I^1 \qquad \dfrac{\dfrac{[A]^1 \quad \dfrac{[A]^1}{A \vee B}\,\vee I_1}{A \wedge (A \vee B)}\,\wedge I}{A \supset A \wedge (A \vee B)}\,\supset I^1$$

Examples

Example 3.9

Prove $\vdash (A \wedge B) \wedge C = A \wedge (B \wedge C)$

$$\dfrac{\dfrac{\dfrac{[(A \wedge B) \wedge C]^1}{A \wedge B}\wedge E_1}{A}\wedge E_1 \qquad \dfrac{\dfrac{\dfrac{[(A \wedge B) \wedge C]^1}{A \wedge B}\wedge E_2}{B} \qquad \dfrac{[(A \wedge B) \wedge C]^1}{C}\wedge E_2}{B \wedge C}\wedge I}{\dfrac{A \wedge (B \wedge C)}{(A \wedge B) \wedge C \supset A \wedge (B \wedge C)}\supset I^1}$$

$$\dfrac{\dfrac{\dfrac{[A \wedge (B \wedge C)]^1}{A}\wedge E_1 \qquad \dfrac{\dfrac{[A \wedge (B \wedge C)]^1}{B \wedge C}\wedge E_2}{B}\wedge E_1}{A \wedge B}\wedge I \qquad \dfrac{\dfrac{[A \wedge (B \wedge C)]^1}{B \wedge C}\wedge E_2}{C}\wedge E_2}{\dfrac{(A \wedge B) \wedge C}{A \wedge (B \wedge C) \supset (A \wedge B) \wedge C}\supset I^1}$$

Examples

Falsity elimination allows to deduce any formula one needs. But falsity always comes from a contradiction.

Example 3.10

Prove $\vdash \neg A \supset (A \supset B)$

$$\dfrac{\dfrac{\dfrac{\dfrac{[\neg A]^1 \quad [A]^2}{\bot}\neg\mathsf{E}}{B}\bot\mathsf{E}}{A \supset B}\supset\mathsf{I}^2}{\neg A \supset (A \supset B)}\supset\mathsf{I}^1$$

Examples

Thinking backwards, not introduction allows to assume the conclusion deprived of the negation.

Example 3.11

Prove $\vdash A \wedge B \supset \neg(A \supset \neg B)$

$$\dfrac{\dfrac{\dfrac{[A \supset \neg B]^1 \qquad \dfrac{[A \wedge B]^2}{A}\wedge E_1}{\neg B}\supset E \qquad \dfrac{[A \wedge B]^2}{B}\wedge E_2}{\bot}\neg E}{\neg(A \supset \neg B)}\neg I^1}{A \wedge B \supset \neg(A \supset \neg B)}\supset I^2$$

Examples

Example 3.12

Prove $\vdash \neg(A \vee B) = \neg A \wedge \neg B$

$$\dfrac{\dfrac{\dfrac{[\neg(A \vee B)]^1 \quad \dfrac{[A]^2}{A \vee B}\vee I_1}{\bot}\neg E}{\neg A}\neg I^2 \quad \dfrac{\dfrac{[\neg(A \vee B)]^1 \quad \dfrac{[B]^3}{A \vee B}\vee I_2}{\bot}\neg E}{\neg B}\neg I^3}{\dfrac{\neg A \wedge \neg B}{\neg(A \vee B) \supset \neg A \wedge \neg B}\supset I^1}\wedge I$$

$$\dfrac{[A \vee B]^1 \quad \dfrac{[A]^2 \quad \dfrac{[\neg A \wedge \neg B]^3}{\neg A}\wedge E_1}{\bot}\neg E \quad \dfrac{[B]^2 \quad \dfrac{[\neg A \wedge \neg B]^3}{\neg B}\wedge E_2}{\bot}\neg E}{\dfrac{\bot}{\neg(A \vee B)}\neg I^1}\vee E^2$$

$$\dfrac{\dfrac{\bot}{\neg(A \vee B)}\neg I^1}{\neg A \wedge \neg B \supset \neg(A \vee B)}\supset I^3$$

Examples

Example 3.13

Prove $\vdash \neg\neg(A \wedge B) \supset \neg\neg A \wedge \neg\neg B$

$$\dfrac{\dfrac{\dfrac{\dfrac{[\neg\neg(A\wedge B)]^{1} \quad \dfrac{\dfrac{\dfrac{[\neg A]^{2} \quad \dfrac{[A\wedge B]^{3}}{A}\wedge E_1}{\bot}\neg E}{\neg(A\wedge B)}\neg I^{3}}{\bot}\neg E}{\neg\neg A}\neg I^{2} \quad \dfrac{\dfrac{\dfrac{[\neg B]^{4} \quad \dfrac{[A\wedge B]^{5}}{B}\wedge E_2}{\bot}\neg E}{\neg(A\wedge B)}\neg I^{5}}{\bot}\neg E}{\neg\neg B}\neg I^{4}}{\neg\neg A\wedge\neg\neg B}\wedge I}{\neg\neg(A\wedge B)\supset\neg\neg A\wedge\neg\neg B}\supset I^{1}$$

Examples

Example 3.14

Prove $\vdash \neg\neg A \wedge \neg\neg B \supset \neg\neg(A \wedge B)$

$$\dfrac{\dfrac{\dfrac{\dfrac{[A]^1 \quad [B]^2}{A \wedge B}\wedge\mathrm{I} \qquad [\neg(A \wedge B)]^3}{\bot}\neg\mathrm{E}}{\neg A}\neg\mathrm{I}^1 \qquad \dfrac{[\neg\neg A \wedge \neg\neg B]^4}{\neg\neg A}\wedge\mathrm{E}_1}{\bot}\neg\mathrm{E} \qquad \dfrac{[\neg\neg A \wedge \neg\neg B]^4}{\neg\neg B}\wedge\mathrm{E}_2}{\bot}\neg\mathrm{E}}{\neg B}\neg\mathrm{I}^2$$

$$\dfrac{\dfrac{\bot}{\neg\neg(A \wedge B)}\neg\mathrm{I}^3}{\neg\neg A \wedge \neg\neg B \supset \neg\neg(A \wedge B)}\supset\mathrm{I}^4$$

Examples

Example 3.15

Prove $\vdash \neg\neg(A \supset B) \supset \neg\neg A \supset \neg\neg B$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[A]^1 \quad [A \supset B]^2}{B}\supset E \quad [\neg B]^3}{\bot}\neg E}{\neg(A \supset B)}\neg I^2 \quad [\neg\neg(A \supset B)]^4}{\bot}\neg E}{\neg A}\neg I^1 \quad [\neg\neg A]^5}{\bot}\neg E}{\neg\neg B}\neg I^3}{\neg\neg A \supset \neg\neg B}\supset I^5}{\neg\neg(A \supset B) \supset \neg\neg A \supset \neg\neg B}\supset I^4$$

Examples

Example 3.16

Prove $\vdash (\neg\neg A \supset \neg\neg B) \supset \neg\neg(A \supset B)$

$$\dfrac{\dfrac{\dfrac{[\neg\neg A \supset \neg\neg B]^{2} \qquad \dfrac{\dfrac{[A]^{3} \quad [\neg A]^{4}}{\bot}\neg E}{\neg\neg A}\supset I^{4}}{\neg\neg B} \supset E \qquad \dfrac{\dfrac{[\neg(A \supset B)]^{1} \quad \dfrac{[B]^{5}}{A \supset B}\supset I}{\bot}\neg E}{\neg B}\neg I^{5}}{\bot}\neg E}{B}\bot E}{A \supset B}\supset I^{3}}{\dfrac{\dfrac{\bot}{\neg\neg(A \supset B)}\neg I^{1}}{(\neg\neg A \supset \neg\neg B) \supset \neg\neg(A \supset B)}\supset I^{2}}\neg E$$

Examples

Proofs involving the Law of Excluded Middle are more difficult. The fundamental strategy is that an application of the principle is required when no other strategy could be applied.

Example 3.17

Prove $\vdash A = \neg\neg A$

$$\dfrac{\dfrac{\dfrac{}{A \vee \neg A}\,\text{lem} \qquad \dfrac{\dfrac{[\neg A]^1 \quad [\neg\neg A]^2}{\bot}\,\neg\text{E}}{A}\,\bot\text{E} \qquad [A]^1}{A}\,\vee\text{E}^1}{\neg\neg A \supset A}\,\supset\text{I}^2 \qquad \dfrac{\dfrac{\dfrac{[\neg A]^1 \quad [A]^2}{\bot}\,\neg\text{E}}{\neg\neg A}\,\neg\text{I}^1}{A \supset \neg\neg A}\,\supset\text{I}^2$$

Examples

Do not rely on the shape of the theorem! Small variations could be provable **without** the Law of Excluded Middle!

Example 3.18

Prove $\vdash \neg A = \neg\neg\neg A$

$$\dfrac{\dfrac{[\neg\neg\neg A]^1 \qquad \dfrac{\dfrac{[\neg A]^2 \quad [A]^3}{\bot}\neg E}{\neg\neg A}\neg I^2}{\neg A}\neg E}{\neg\neg\neg A \supset \neg A}\neg I^3 \supset I^1 \qquad \dfrac{\dfrac{[\neg\neg A]^1 \quad [\neg A]^2}{\bot}\neg E}{\neg\neg\neg A}\neg I^1 \supset I^2$$

Examples

You may think the Law of Excluded Middle is about negation. This is false: there are elementary facts in which negation does not appear, which **require** the Law of Excluded Middle to be proved.

Example 3.19

Prove $\vdash (A \supset B) \vee (B \supset A)$

$$\dfrac{\dfrac{}{A \vee \neg A}\,\text{lem} \quad \dfrac{\dfrac{[A]^1}{B \supset A}\,\supset\text{I}}{(A \supset B) \vee (B \supset A)}\,\vee\text{I}_2 \quad \dfrac{\dfrac{\dfrac{[A]^2 \quad [\neg A]^1}{\bot}\,\neg\text{E}}{B}\,\bot\text{E}}{A \supset B}\,\supset\text{I}^2}{(A \supset B) \vee (B \supset A)}\,\vee\text{I}_1 \qquad \vee\text{E}^1$$

Examples

Example 3.20

Prove $\vdash ((A \supset B) \supset A) \supset A$

$$\dfrac{\dfrac{\dfrac{}{A \vee \neg A}\,\text{lem} \quad \dfrac{\dfrac{[\neg A]^1 \quad [A]^3}{\bot}\,\neg\text{E}}{B}\,\bot\text{E}}{A}\,\vee\text{E}^1 \quad \dfrac{[(A \supset B) \supset A]^2}{A}\,\supset\text{E}}{((A \supset B) \supset A) \supset A}\,\supset\text{I}^2$$

Examples

Example 3.21

Prove $\vdash A \supset B = \neg B \supset \neg A$

$$\dfrac{\dfrac{\dfrac{[A \supset B]^1 \quad [A]^2}{B}\supset E \qquad [\neg B]^3}{\bot}\neg E}{\dfrac{\neg A}{\neg B \supset \neg A}\supset I^3}\neg I^2 \Big/ (A \supset B) \supset (\neg B \supset \neg A)\ \supset I^1$$

$$\dfrac{\dfrac{\dfrac{[\neg B \supset \neg A]^2 \quad [\neg B]^1}{\neg A}\supset E \qquad [A]^3}{\bot}\neg E}{\dfrac{\dfrac{}{B \vee \neg B}\,\text{lem} \qquad [B]^1}{B}\vee E^1}\bot E \Big/ A \supset B\ \supset I^3 \Big/ (\neg B \supset \neg A) \supset (A \supset B)\ \supset I^2$$

Examples

Example 3.22

Prove $\vdash A \supset B = \neg(A \wedge \neg B)$

$$\dfrac{\dfrac{\dfrac{\dfrac{[A \supset B]^1 \quad \dfrac{\dfrac{[A \wedge \neg B]^2}{A}\wedge E_1 \quad \dfrac{[A \wedge \neg B]^2}{\neg B}\wedge E_2}{B}\supset E}{\bot}\neg E}{\neg(A \wedge \neg B)}\neg I^2}{(A \supset B) \supset \neg(A \wedge \neg B)}\supset I^1 \qquad \dfrac{\dfrac{\dfrac{\dfrac{[A]^2 \quad [\neg B]^1}{A \wedge \neg B}\wedge I \quad [\neg(A \wedge \neg B)]^3}{\bot}\neg E}{B}\bot E \quad \dfrac{}{B \vee \neg B}\text{lem} \quad [B]^1}{B}\vee E^1}{A \supset B}\supset I^2}{\neg(A \wedge \neg B) \supset (A \supset B)}\supset I^3$$

Examples

Example 3.23

Prove $\vdash A \vee B = \neg A \supset B$

$$\dfrac{\dfrac{\dfrac{[A \vee B]^1 \qquad \dfrac{\dfrac{[A]^2 \quad [\neg A]^3}{\bot}\neg E}{B}\bot E \qquad [B]^2}{B}\vee E^2}{\neg A \supset B}\supset I^3}{A \vee B \supset (\neg A \supset B)}\supset I^1$$

$$\dfrac{\dfrac{\dfrac{}{A \vee \neg A}\,\text{lem} \qquad \dfrac{[A]^1}{A \vee B}\vee I_1 \qquad \dfrac{\dfrac{[\neg A]^1 \quad [\neg A \supset B]^2}{B}\supset E}{A \vee B}\vee I_2}{A \vee B}\vee E^1}{(\neg A \supset B) \supset A \vee B}\supset I^2$$

References

Exercises could be find in any standard textbook, see, e.g., Chapter 1 of *John Bell* and *Moshé Machover*, A Course in Mathematical Logic, North-Holland, (1977), ISBN 0-7204-28440.

Proving techniques come from the completeness proof.

Mathematical Logic

Lecture 4

Dr Marco Benini

marco.benini@uninsubria.it

Dipartimento di Scienza e Alta Tecnologia
Università degli Studi dell'Insubria

a.a. 2016/17

Syllabus

Propositional logic:

- Semantics: truth tables
- Examples
- Applications
- Soundness

Semantics

The intended meaning of propositional logic can be formalised. In this way, we will get a first, very simple semantics for the syntax introduced in the previous lessons.

Definition 4.1 (Truth-tables semantics)

Fixed a map $v\colon V \to \{0,1\}$ from the set of variables V to the truth values, denoted by 0 and 1, the *meaning* $[\![A]\!]$ of a formula A is inductively defined as follows:

- if $A \in V$ is a variable, then $[\![A]\!] = v(A)$;
- $[\![\top]\!] = 1$;
- $[\![\bot]\!] = 0$;

↪

Semantics

↪ (Truth-tables semantics)

- if $A \equiv B \wedge C$ then $[\![A]\!]$ is calculated according to

$[\![B]\!]$	$[\![C]\!]$	$[\![B \wedge C]\!]$
0	0	0
0	1	0
1	0	0
1	1	1

- if $A \equiv B \vee C$ then $[\![A]\!]$ is calculated according to

$[\![B]\!]$	$[\![C]\!]$	$[\![B \vee C]\!]$
0	0	0
0	1	1
1	0	1
1	1	1

↪

Semantics

↪ (Truth-tables semantics)

- if $A \equiv \neg B$ then $[\![A]\!]$ is calculated according to

$[\![B]\!]$	$[\![\neg B]\!]$
0	1
1	0

- if $A \equiv B \supset C$ then $[\![A]\!]$ is calculated according to

$[\![B]\!]$	$[\![C]\!]$	$[\![B \supset C]\!]$
0	0	1
0	1	1
1	0	0
1	1	1

Example 4.2

We can show that the formula $x \wedge y \supset x \vee y$ is true whatever values we may assign to x and y;

$[\![x]\!]$	$[\![y]\!]$	$[\![x \wedge y]\!]$	$[\![x \vee y]\!]$	$[\![x \wedge y \supset x \vee y]\!]$
0	0	0	0	1
0	1	0	1	1
1	0	0	1	1
1	1	1	1	1

The corresponding proofs in natural deduction are:

$$\dfrac{\dfrac{\dfrac{[x \wedge y]^*}{x}\wedge E_1}{x \vee y}\vee I_1}{x \wedge y \supset x \vee y}\supset I^* \qquad \dfrac{\dfrac{\dfrac{[x \wedge y]^*}{y}\wedge E_2}{x \vee y}\vee I_2}{x \wedge y \supset x \vee y}\supset I^*$$

Applications

Truth tables are widely used in the synthesis of (logical) circuits, and many techniques to minimise the number of electronic gates, each one implementing a logical connective, have been implemented.

In logic, truth tables are not an effective way to check whether a formula is true for any assignment of its variables: the number of assignment to try is 2^n, with n the number of variables, so it grows exponentially with respect to the number of variables.

Anyway, in pure logic, truth tables are a very effective way to construct a minimal set of connectives. In fact, connectives are not independent, as they can be mutually defined.

Interdependence of connectives

Proposition 4.3

Negation can be defined using implication and falsity.

Proof.

Checking the truth tables, one immediately realises that $\neg A$ is equivalent to $A \supset \bot$. □

Proposition 4.4

The set of connectives $\wedge$, $\vee$, and $\neg$ suffices to define all the others.

Proof.

Just checking the truth tables, one can see that

- $\top$ can be defined as $\neg X \vee X$, for any choice of X;
- $\bot$ can be defined as $\neg \top$;
- $A \supset B$ can be defined as $\neg A \vee B$. □

Interdependence of connectives

Proposition 4.5

Conjunction can be defined from disjunction and negation. Also, disjunction can be defined from conjunction and negation.

Proof.

Checking the proof table, it is immediate to see that

- $A \wedge B$ is the same as $\neg(\neg A \vee \neg B)$;
- $A \vee B$ is the same as $\neg(\neg A \wedge \neg B)$. □

Usually, $\neg(A \wedge B) = \neg A \vee \neg B$ and $\neg(A \vee B) = \neg A \wedge \neg B$ are referred to as the De Morgan's Laws. Here, $A = B$ between two formulae A and B means that both $A \supset B$ and $B \supset A$ hold, i.e., A and B are equivalent.

Interdependence of connectives

So, the following set of connectives are sufficient to define all the others:

- $\{\bot, \supset\}$;
- $\{\neg, \wedge\}$;
- $\{\neg, \vee\}$;
- $\{\neg, \supset\}$.

But, in principle, one can reduce to a single connective, although this is impractical. Define $A \mid B = \neg(A \wedge B)$, which is known as *Sheffer's stroke*. Then, using the truth tables it is easy to prove

- $\neg A = A \mid A$;
- $A \supset B = A \mid (B \mid B)$.

Soundness

We want to show that every conclusion we may derive in the proof system is true whenever all the assumptions it depends upon are true.

Before stating the theorem and proving it, we should make one important remark. The collection of proofs is inductively generated by the inference rules. So, we can reason about a provable statement by saying: if A is provable, let π be a proof of A. If a property holds for every proof, then it holds for π, too.

To prove that a property holds for every proof, we can prove that each inference rule *preserves* the property, which means that, assuming the property to hold for the proofs in the premises of the rule, we have to show that the proof whose last rule is the inference rule under examination, has the property, too. In the case of the Soundness Theorem, the property of interest is 'the conclusion is true'.

Soundness

Theorem 4.6 (Soundness)

If Γ is a set of formulae, and we have a proof $\pi : \Gamma \vdash A$ in the natural deduction system, then whenever each formula in Γ is true, so is A.

Proof. (i)

The main hypothesis is that, for every $G \in \Gamma$, $[\![G]\!] = 1$. We proceed by induction on the definition of the proof π, showing that if all the antecedents of an inference rules satisfy the property in the statement, so does the conclusion:

- if π is an instance of the assumption rule, then $A \in \Gamma$, so $[\![A]\!] = 1$ by hypothesis.
- if π is an instance of the $\top$I rule, then $A \equiv \top$, so $[\![A]\!] = 1$. ↪

Soundness

↪ Proof. (ii)

- if π is an instance of the $\bot$E rule, then, by induction hypothesis, $[\![\bot]\!] = 1$, but we know that $[\![\bot]\!] = 0$, thus $0 = 1$. Then, since $[\![A]\!] \in \{0, 1\}$, it follows that $[\![A]\!] = 1$.
- if π is an instance of the Law of Excluded Middle, $A \equiv B \vee \neg B$. But $[\![B \vee \neg B]\!] = 1$, as it is immediate to check by the truth tables.
- if π is an instance of $\neg$I, then, by the induction hypothesis applied to $\pi' : \Gamma \cup \{A\} \vdash \bot$, we have that $[\![A]\!] = 1$ implies $[\![\bot]\!] = 1$. Then, the contrapositive form of the implication says that $[\![\bot]\!] \neq 1$ implies $[\![A]\!] \neq 1$, which means $[\![\bot]\!] = 0$ implies $[\![A]\!] = 0$. But we know that $[\![\bot]\!] = 0$, so $[\![A]\!] = 0$, that is $[\![\neg A]\!] = 1$.
- if π is an instance of $\neg$E, then, by the induction hypothesis applied twice to both antecedents, we get that $[\![\neg A]\!] = 1$ and $[\![A]\!] = 1$. Thus, $0 = [\![A]\!] = 1$. Then $[\![\bot]\!] = 0 = 1$. ↪

Soundness

↪ Proof. (iii)

- if π is an instance of $\wedge\mathsf{I}$, then, $A \equiv B \wedge C$ and, by the induction hypothesis applied to both antecedents, $[\![B]\!] = 1$ and $[\![C]\!] = 1$. So, by the truth table of conjunction, $[\![B \wedge C]\!] = 1$.
- if π is an instance of $\wedge\mathsf{E}_1$, then the antecedent is a proof of $A \wedge B$ from Γ. Applying the induction hypothesis, we get that $[\![A \wedge B]\!] = 1$, so, by the truth table of conjunction, we derive that $[\![A]\!] = 1$.
- if π is an instance of $\wedge\mathsf{E}_2$, then the antecedent is a proof of $B \wedge A$ from Γ. Applying the induction hypothesis, we get that $[\![B \wedge A]\!] = 1$, so, by the truth table of conjunction, we derive that $[\![A]\!] = 1$. ↪

Soundness

↪ Proof. (iv)

- if π is an instance of $\vee \mathrm{I}_1$ then, $A \equiv B \vee C$ and the antecedent is a proof of B from Γ. By the induction hypothesis, $[\![B]\!] = 1$, so, by the truth table of disjunction, $[\![B \vee C]\!] = 1$.
- if π is an instance of $\vee \mathrm{I}_2$ then, $A \equiv B \vee C$ and the antecedent is a proof of C from Γ. By the induction hypothesis, $[\![C]\!] = 1$, so, by the truth table of disjunction, $[\![B \vee C]\!] = 1$.
- if π is an instance of $\vee \mathrm{E}$ then, applying the induction hypothesis to the first antecedent, we get that $[\![B \vee C]\!] = 1$ for appropriate B and C. Thus, by the truth table of disjunction, $[\![B]\!] = 1$, or $[\![C]\!] = 1$. In the former case, applying the induction hypothesis to the second antecedent, we get that $[\![A]\!] = 1$. In the latter case, applying the induction hypothesis to the third antecedent, we get that $[\![A]\!] = 1$. ↪

Soundness

↪ Proof. (v)

- if π is an instance of $\supset$I, then $A \equiv B \supset C$. If $[\![B]\!] = 0$ then, by the truth table of implication, $[\![B \supset C]\!] = 1$. Otherwise, $[\![B]\!] = 1$, and we can apply the induction hypothesis to the antecedent of the inference rule, obtaining that $[\![C]\!] = 1$. Thus, by the truth table of implication, $[\![B \supset C]\!] = 1$.
- if π is an instance of $\supset$E, then, applying the induction hypothesis to both antecedents, we get $[\![B \supset A]\!] = 1$ and $[\![B]\!] = 1$. Thus, by the truth table of implication, it follows that $[\![A]\!] = 1$, too. □

References

The truth table semantics is described in Section 2.4 of the lecture notes.

The soundness theorem is folklore. In fact, we will see soon a more interesting and powerful version of it, which uses a more refined semantics.

The interest of the soundness theorem lies in the structure of its proof: most soundness theorems are proved by induction on the structure of proofs, checking that each inference rule preserves the truth of antecedents into the consequence. It is important to become acquainted with this technique.

Mathematical Logic

Lecture 5

Dr Marco Benini

marco.benini@uninsubria.it

Dipartimento di Scienza e Alta Tecnologia
Università degli Studi dell'Insubria

a.a. 2016/17

Syllabus

Propositional logic:

- Orders
- Lattices
- Boolean algebras
- Semantics
- Soundness

Orders

A rather more interesting semantics for propositional logic comes from the algebra of orders. In the following, we will develop what is needed to introduce it.

Definition 5.1 (Order)

An *order* $\mathcal{O} = \langle S; \leq \rangle$ is a set S equipped with a binary relation $\leq$ which is

- *reflexive*, i.e., for all $x \in S$, $x \leq x$;
- *anti-symmetric*, i.e., for all $x, y \in S$, when $x \leq y$ and $y \leq x$, then $x = y$;
- *transitive*, i.e., for all $x, y, z \in S$, if $x \leq y$ and $y \leq z$, then $x \leq z$.

Orders

Definition 5.2 (Least upper bound)

Fixed an order $\mathscr{O} = \langle S; \leq \rangle$ and a $U \subseteq S$, we call the element $m \in S$, if it exists, the *least upper bound* (lub), or *supremum*, or *join*, of U whenever

- for every $x \in U$, $x \leq m$;
- for each $w \in S$ such that, for every $x \in U$, $x \leq w$, it holds that $m \leq w$.

Definition 5.3 (Greatest lower bound)

Fixed an order $\mathscr{O} = \langle S; \leq \rangle$ and a $U \subseteq S$, we call the element $m \in S$, if it exists, the *greatest lower bound* (glb), or *infimum*, or *meet* of U whenever

- for every $x \in U$, $m \leq x$;
- for each $w \in S$ such that, for every $x \in U$, $w \leq x$, it holds that $w \leq m$.

Lattices

Definition 5.4 (Lattice)

An order $\mathscr{O} = \langle S; \leq \rangle$ is called a *lattice* when, for every pair $x, y \in S$, there exists the join of $\{x, y\}$, denoted by $x \vee y$, and there exists the meet of $\{x, y\}$, denoted by $x \wedge y$.
Moreover, a lattice is said to be *bounded* when, for every finite $U \subseteq S$, there is $\bigvee U$, the join of U, and $\bigwedge U$, the meet of U. Conventionally, $\bigvee \emptyset$ is denoted by $\bot$, and $\bigwedge \emptyset$ is denoted by $\top$.

Lattices

Proposition 5.5

In a bounded lattice $\langle S; \leq \rangle$, every element is greater that $\bot$ and less than $\top$.

Proof.

Since $\top = \bigwedge \emptyset$, by definition of meet, for all $x \in \emptyset$, $\top \leq x$, and, for any $y \in S$ such that for all $x \in \emptyset$, $y \leq x$, it holds that $y \leq \top$. But there are no elements in $\emptyset$, so $y \leq \top$ for any $y \in S$.

Since $\bot = \bigvee \emptyset$, by definition of join, for all $x \in \emptyset$, $x \leq \bot$, and, for any $y \in S$ such that for all $x \in \emptyset$, $x \leq y$, it holds that $\bot \leq y$. But there are no elements in $\emptyset$, so $\bot \leq y$ for any $y \in S$. □

Lattices

Proposition 5.6

In a bounded lattice $\langle S; \leq \rangle$, $\bigvee S = \top$ *and* $\bigwedge S = \bot$.

Proof.

By definition of join, for every $x \in S$, $x \leq \bigvee S$, and, by Proposition 5.5, $\top$ is such that, for all $x \in S$, $x \leq \top$. So, $\top \leq \bigvee S$ and $\bigvee S \leq \top$. By anti-symmetry, $\bigvee S = \top$.

By definition of meet, for every $x \in S$, $\bigwedge S \leq x$, and, by Proposition 5.5, $\bot$ is such that, for all $x \in S$, $\bot \leq x$. So, $\bot \leq \bigwedge S$ and $\bigwedge S \leq \bot$. By anti-symmetry, $\bigwedge S = \bot$. □

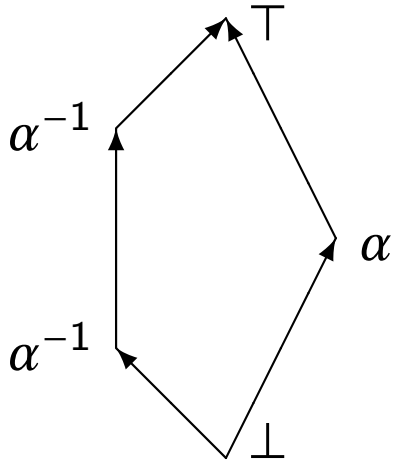
Lattices

Definition 5.7 (Complemented lattice)

A bounded lattice $\mathscr{O} = \langle S; \leq \rangle$ is said to be *complemented* when, for each element $x \in S$, there is an element $y \in S$ such that

- $x \wedge y = \bot$;
- $x \vee y = \top$.

The element y is not necessarily unique. For example



Lattices

Definition 5.8 (Distributive lattice)

A lattice $\mathscr{O} = \langle S; \leq \rangle$ is said to be *distributive* when, for every $x, y, z \in S$, $x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$.

Proposition 5.9

In every lattice, $x \wedge y = y \wedge x$, *and* $x \vee y = y \vee x$.

Proof.

Immediate, by definition of meet and join. □

Lattices

Proposition 5.10

For each x in a bounded lattice, $x = x \wedge \top$ and $x = x \vee \bot$

Proof.

Immediate, by definition of meet and join, and Proposition 5.5. □

Proposition 5.11

For each x and y in a lattice, $x \vee (x \wedge y) = x$

Proof.

By definition of meet, $x \leq x \vee (x \wedge y)$, so it suffices to show $x \vee (x \wedge y) \leq x$.

But $x \leq x$ by reflexivity, and $x \wedge y \leq x$ by definition of join, so $x \vee (x \wedge y) \leq x$ by definition of meet. □

Lattices

Proposition 5.12

In any lattice, if $x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$ *then* $x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$.

Proof.

By hypothesis, $(x \vee y) \wedge (x \vee z) = ((x \vee y) \wedge x) \vee ((x \vee y) \wedge z)$, but $\wedge$ is commutative, so we can apply the hypothesis twice inside the brackets, obtaining $(x \wedge x) \vee (x \wedge y) \vee (x \wedge z) \vee (y \wedge z)$.

Thus $(x \vee y) \wedge (x \vee z) = x \vee (x \wedge y) \vee (x \wedge z) \vee (y \wedge z)$. Grouping the first two element on the right-hand side, we get $(x \vee y) \wedge (x \vee z) = (x \vee (x \wedge y)) \vee (x \wedge z) \vee (y \wedge z)$.

But the first element of the right-hand side reduces to x, so $(x \vee y) \wedge (x \vee z) = (x \vee (x \wedge z)) \vee (y \wedge z)$. Reducing again the first element, it follows that $(x \vee y) \wedge (x \vee z) = x \vee (y \wedge z)$. □

Lattices

Proposition 5.13

In any bounded distributive complemented lattice, each element x has a unique complement, denoted by $\neg x$.

Proof.

Suppose the element x has two complements y and z. Then, by definition of complement

- $x \wedge y = \bot = x \wedge z$,
- $x \vee y = \top = x \vee z$.

Thus, $y = y \wedge \top = y \wedge (x \vee z) = (y \wedge x) \vee (y \wedge z) = \bot \vee (y \wedge z) = (z \wedge x) \vee (z \wedge y) = z \wedge (x \vee y) = z \wedge \top = z$. □

Boolean algebras

Definition 5.14 (Boolean algebra)

A *Boolean algebra* is a bounded distributive complemented lattice.

Example 5.15

The set $\{0,1\}$, with the ordering $0 \leq 1$, is a Boolean algebra, with $\top = 1$ and $\bot = 0$. This is the structure supporting the truth-table semantics.

Boolean algebras

Example 5.16

Fixed a set U, the powerset $\wp(U) = \{S : S \subseteq U\}$ ordered by inclusion, is a Boolean algebra. The complement of S is the difference $U \setminus S$, while $\wedge$ is the intersection, and $\vee$ is the union.

Example 5.17

Let $n \in \mathbb{N}$ be such that it cannot be divided by the square of any other number, e.g., $105 = 3 \cdot 5 \cdot 7$. Then, the divisors of n form a Boolean algebra, with the operations of greatest common divisor, least common multiple, and the complement of x being n/x.

Semantics

We introduced Boolean algebra for a precise purpose: interpreting propositional logic.

Definition 5.18 (Semantics)

Fixed a Boolean algebra $\mathscr{O} = \langle O; \leq \rangle$, and $\nu \colon V \to O$ mapping each variable into an element of the algebra, the interpretation $[\![A]\!]$ of a formula A is inductively defined as:

- if A is a variable, $[\![A]\!] = \nu(A)$;
- if $A \equiv \top$, $[\![A]\!] = \top$, the maximum element of $\mathscr{O}$;
- if $A \equiv \bot$, $[\![A]\!] = \bot$, the minimum element of $\mathscr{O}$;
- if $A \equiv B \wedge C$, $[\![A]\!] = [\![B]\!] \wedge [\![C]\!]$, the meet of the interpretations of conjuncts;
- if $A \equiv B \vee C$, $[\![A]\!] = [\![B]\!] \vee [\![C]\!]$, the join of the interpretations of disjuncts;
- if $A \equiv B \supset C$, $[\![A]\!] = \neg [\![B]\!] \vee [\![C]\!]$, that is $[\![A]\!] = [\![\neg B \vee C]\!]$, interpreting implication as a *relative complement*;
- if $A \equiv \neg B$, $[\![A]\!] = \neg [\![B]\!]$, the complement of the interpretation of B.

Examples

Example 5.19

Let us fix the Boolean algebra given by the powerset of $\mathbb{N}$, ordered by inclusion. For simplicity, the variables have the form x_n, with $n \in \mathbb{N}$, and $\nu(x_n) = \{n\}$. It is immediate to check that meets are unions, and joins are intersections. Also, $\bot = \emptyset$ and $\top = \mathbb{N}$.

Then, $[\![x_3 \vee \neg x_3]\!] = [\![x_3]\!] \cup (\mathbb{N} \setminus [\![x_3]\!]) = \{3\} \cup (\mathbb{N} \setminus \{3\}) = \mathbb{N}$.

Also, $[\![x_5 \wedge \neg x_5]\!] = [\![x_5]\!] \cap (\mathbb{N} \setminus [\![x_5]\!]) = \{5\} \cap (\mathbb{N} \setminus \{5\}) = \emptyset$.

Finally, $[\![x_3 \vee \neg x_5]\!] = [\![x_3]\!] \cup (\mathbb{N} \setminus [\![x_5]\!]) = \{3\} \cup (\mathbb{N} \setminus \{5\}) = \mathbb{N} \setminus \{5\}$.

Every 'true' formula seems to be intepreted in the top element of the algebra; every 'false' formula seems to be interpreted in the bottom element of the algebra.

But a formula, which, according to the truth table semantics, is sometimes 'true' and sometimes 'false', depending on the values of its variables, seems to be interpreted in a 'truth-value' which is neither $\top$ nor $\bot$.

Soundness

Definition 5.20 (Validity)

A formula A is *valid* or *true* in a Boolean algebra $\mathscr{O} = \langle O; \leq \rangle$ together with an interpretation $\nu \colon V \to O$ of variables, when $[\![A]\!] = \top$.
A set of formulae is *valid* or *true* when each formula in the set is valid.

Theorem 5.21 (Soundness)

In any Boolean algebra $\mathscr{O} = \langle O; \leq \rangle$, for any interpretation $\nu \colon V \to O$ of variables, which makes true the theory T and the assumptions in the finite set Δ, if A is the conclusion of a proof π from Δ in T, then A is valid.

Soundness

Proof. (i)

The proof is by induction on the structure of π: we show that the interpretation of the conclusion A is greater than $\bigwedge_{G\in\Gamma}[\![G]\!]$, with Γ the finite set of assumptions occurring in the proof of A:

- if π is a proof by assumption, then $A\in\Gamma$ and, by definition of $\wedge$, $\bigwedge_{G\in\Gamma}[\![G]\!]\leq[\![A]\!]$.
- if π is a proof by axiom, then $A\in T$, and, by hypothesis, $[\![A]\!]=\top$, so $\bigwedge_{G\in\Gamma}[\![G]\!]\leq[\![A]\!]$ by definition of $\top$.
- if π is an instance of the Law of Excluded Middle, then $A\equiv B\vee\neg B$, and $[\![A]\!]=[\![B\vee\neg B]\!]=[\![B]\!]\vee\neg[\![B]\!]=\top$ by definition of complement in a Boolean algebra. Thus $\bigwedge_{G\in\Gamma}[\![G]\!]\leq[\![A]\!]=\top$ by definition of $\top$.
- if π is an instance of $\top$-introduction, then $A\equiv\top$, so $[\![A]\!]=[\![\top]\!]=\top$. Thus $\bigwedge_{G\in\Gamma}[\![G]\!]\leq[\![A]\!]=\top$ by definition of $\top$.

↪

Soundness

↪ Proof. (ii)

- if π is an instance of $\bot$-elimination, then, by induction hypothesis, $\bot \leq \bigwedge_{G\in\Gamma}[\![G]\!] \leq [\![\bot]\!] = \bot$. Thus, by anti-symmetry, $\bigwedge_{G\in\Gamma}[\![G]\!] = \bot$. So, by definition of $\bot$, $\bot = \bigwedge_{G\in\Gamma}[\![G]\!] \leq [\![A[\![$.
- if π is an instance of $\wedge$-introduction, then $A \equiv B \wedge C$, and by induction hypothesis twice, $\bigwedge_{G\in\Gamma}[\![G]\!] \leq [\![B]\!]$ and $\bigwedge_{G\in\Gamma}[\![G]\!] \leq [\![C]\!]$. Thus, by definition of $\wedge$, $\bigwedge_{G\in\Gamma}[\![G]\!] \leq [\![B]\!] \wedge [\![C]\!] = [\![B \wedge C]\!] = [\![A]\!]$.
- if π is an instance of $\wedge_1$-elimination, then, by induction hypothesis, for some formula B, $\bigwedge_{G\in\Gamma}[\![G]\!] \leq [\![A \wedge B]\!] = [\![A]\!] \wedge [\![B]\!]$. Thus, by definition of $\wedge$, $\bigwedge_{G\in\Gamma}[\![G]\!] \leq [\![A]\!]$.
- if π is an instance of $\wedge_2$-elimination, then, by induction hypothesis, for some formula B, $\bigwedge_{G\in\Gamma}[\![G]\!] \leq [\![B \wedge A]\!] = [\![B]\!] \wedge [\![A]\!]$. Thus, by definition of $\wedge$, $\bigwedge_{G\in\Gamma}[\![G]\!] \leq [\![A]\!]$.

↪

Soundness

↪ Proof. (iii)

- if π is an instance of $\vee_1$-introduction, then $A \equiv B \vee C$ and, by induction hypothesis, $\bigwedge_{G \in \Gamma} [\![G]\!] \leq [\![B]\!]$. So, by definition of $\vee$, $\bigwedge_{G \in \Gamma} [\![G]\!] \leq [\![B]\!] \leq [\![B]\!] \vee [\![C]\!] = [\![B \vee C]\!] = [\![A]\!]$.
- if π is an instance of $\vee_2$-introduction, then $A \equiv B \vee C$ and, by induction hypothesis, $\bigwedge_{G \in \Gamma} [\![G]\!] \leq [\![C]\!]$. So, by definition of $\vee$, $\bigwedge_{G \in \Gamma} [\![G]\!] \leq [\![C]\!] \leq [\![B]\!] \vee [\![C]\!] = [\![B \vee C]\!] = [\![A]\!]$.
- if π is an instance of $\vee$-elimination, then, by induction hypothesis, for some formulae B and C, $\bigwedge_{G \in \Gamma} [\![G]\!] \leq [\![B \vee C]\!] = [\![B]\!] \vee [\![C]\!]$, $[\![B]\!] \wedge \bigwedge_{G \in \Gamma} [\![G]\!] \leq [\![A]\!]$, and $[\![C]\!] \wedge \bigwedge_{G \in \Gamma} [\![G]\!] \leq [\![A]\!]$. It follows that, by definition of $\vee$ and distributing, $([\![B]\!] \wedge \bigwedge_{G \in \Gamma} [\![G]\!]) \vee ([\![C]\!] \wedge \bigwedge_{G \in \Gamma} [\![G]\!]) = ([\![B]\!] \vee [\![C]\!]) \wedge \bigwedge_{G \in \Gamma} [\![G]\!] \leq [\![A]\!]$. But, since $\bigwedge_{G \in \Gamma} [\![G]\!] \leq [\![B]\!] \vee [\![C]\!]$, $([\![B]\!] \vee [\![C]\!]) \wedge \bigwedge_{G \in \Gamma} [\![G]\!] = \bigwedge_{G \in \Gamma} [\![G]\!]$ by definition of $\wedge$, so $\bigwedge_{G \in \Gamma} [\![G]\!] \leq [\![A]\!]$.

↪

Soundness

↪ Proof. (iv)

- if π is an instance of $\supset$-introduction, then $A \equiv B \supset C$ for some formulae B and C. By induction hypothesis, $[\![B]\!] \wedge \bigwedge_{G \in \Gamma} [\![G]\!] \leq [\![C]\!]$. So, by definition of $\vee$, $[\![B]\!] \wedge \bigwedge_{G \in \Gamma} [\![G]\!] \leq \neg [\![B]\!] \vee [\![C]\!]$. Evidently, $\neg [\![B]\!] \leq \neg [\![B]\!] \vee [\![C]\!]$. Thus, by definition of $\vee$, $[\![A]\!] = [\![B \supset C]\!] = \neg [\![B]\!] \vee [\![C]\!] \geq \neg [\![B]\!] \vee ([\![B]\!] \wedge \bigwedge_{G \in \Gamma} [\![G]\!])$. Distributing and by definition of complement, $[\![A]\!] \geq (\neg [\![B]\!] \vee [\![B]\!]) \wedge (\neg [\![B]\!] \vee \bigwedge_{G \in \Gamma} [\![G]\!]) = \top \wedge (\neg [\![B]\!] \vee \bigwedge_{G \in \Gamma} [\![G]\!]) = \neg [\![B]\!] \vee \bigwedge_{G \in \Gamma} [\![G]\!]$. By definition of $\vee$, $\bigwedge_{G \in \Gamma} [\![G]\!] \leq \neg [\![B]\!] \vee \bigwedge_{G \in \Gamma} [\![G]\!] \leq [\![A]\!]$.
- if π is an instance of $\supset$-elimination, then, for some formula B, by induction hypothesis twice, $\bigwedge_{G \in \Gamma} [\![G]\!] \leq [\![B \supset A]\!]$ and $\bigwedge_{G \in \Gamma} [\![G]\!] \leq [\![B]\!]$. By definition of $\wedge$, $\bigwedge_{G \in \Gamma} [\![G]\!] \leq [\![B \supset A]\!] \wedge [\![B]\!]$. But $[\![B \supset A]\!] = \neg [\![B]\!] \vee [\![A]\!]$. So, $\bigwedge_{G \in \Gamma} [\![G]\!] \leq (\neg [\![B]\!] \vee [\![A]\!]) \wedge [\![B]\!]$. Distributing and by definition of $\neg$, $\bigwedge_{G \in \Gamma} [\![G]\!] \leq (\neg [\![B]\!] \wedge [\![B]\!]) \vee ([\![A]\!] \wedge [\![B]\!]) = \bot \vee ([\![A]\!] \wedge [\![B]\!]) = [\![A]\!] \wedge [\![B]\!] \leq [\![A]\!]$.

↪

Soundness

↪ Proof. (v)

- if π is an instance of $\neg$-introduction, then $A \equiv \neg B$ for some formula B. So, by induction hypothesis, $[\![B]\!] \wedge \bigwedge_{G \in \Gamma} [\![G]\!] \leq [\![\bot]\!] = \bot$. Thus, by definition of $\bot$ and anti-symmetry, $[\![B]\!] \wedge \bigwedge_{G \in \Gamma} [\![G]\!] = \bot$. Then, $[\![A]\!] = [\![\neg B]\!] = \neg [\![B]\!] = \neg [\![B]\!] \vee \bot = \neg [\![B]\!] \vee ([\![B]\!] \wedge \bigwedge_{G \in \Gamma} [\![G]\!])$, and, distributing, $[\![A]\!] = (\neg [\![B]\!] \vee [\![B]\!]) \wedge (\neg [\![B]\!] \vee \bigwedge_{G \in \Gamma} [\![G]\!]) = \top \wedge ([\![\neg B]\!] \vee \bigwedge_{G \in \Gamma} [\![G]\!]) = [\![A]\!] \vee \bigwedge_{G \in \Gamma} [\![G]\!]$. Thus, by definition of $\vee$, $\bigwedge_{G \in \Gamma} [\![G]\!] \leq [\![A]\!] \vee \bigwedge_{G \in \Gamma} [\![G]\!] = [\![A]\!]$.
- if π is an instance of $\neg$-elimination, then $A \equiv \bot$ and, by induction hypothesis twice, $\bigwedge_{G \in \Gamma} [\![G]\!] \leq [\![\neg B]\!]$ and $\bigwedge_{G \in \Gamma} [\![G]\!] \leq [\![B]\!]$. But $[\![\neg B]\!] = \neg [\![B]\!]$. So, by definition of $\wedge$, $\bigwedge_{G \in \Gamma} [\![G]\!] \leq \neg [\![B]\!] \wedge [\![B]\!]$. By definition of complement, $\bigwedge_{G \in \Gamma} [\![G]\!] \leq \neg [\![B]\!] \wedge [\![B]\!] = \bot = [\![A]\!]$.

Hence, for every formula A being the conclusion of a proof from Δ in the theory T, $\bigwedge_{G \in \Delta} [\![G]\!] \leq [\![A]\!]$. But, by hypothesis, for every $G \in \Delta$, $[\![G]\!] = \top$, so $\bigwedge_{G \in \Delta} [\![G]\!] = \top$, thus, by definition of $\top$, $\top \leq [\![A]\!] \leq \top$, that is, by anti-symmetry, $[\![A]\!] = \top$. □

References

Boolean algebras, in the form of the powerset of a set, have been introduced for the first time in *George Boole*, An Investigation of the Laws of Thought, Prometheus Books, (2003), reprint from the original edition (1854), ISBN 978-1-59102-089-9.

Two excellent references for orders, lattices, and Boolean algebras are *B. A. Davey and H. A. Priestley*, Introduction to Lattices and Order, Cambridge University Press, (2002), ISBN 978-0-521-78451-1, and *George Grätzer*, General Lattice Theory, second edition, Birkhäuser, (1996), ISBN 978-3-7643-6996-5.

The idea of the proof of the Soundness Theorem is folklore: in fact, the proof itself is adapted from a more general result which uses the internal logic of a Boolean topos. This is an advanced topic, which will not be covered in the course, and the interested student can give a glimpse to *P. Johnstone*, Sketches of an Elephant: A Topos Theory Compendium, two volumes, Oxford University Press (2002), ISBN 978-0-19-853425-9 and 978-0-19-851598-2.

Mathematical Logic

Lecture 6

Dr Marco Benini

marco.benini@uninsubria.it

Dipartimento di Scienza e Alta Tecnologia
Università degli Studi dell'Insubria

a.a. 2016/17

Syllabus

Propositional logic:

- Completeness

Completeness

We will show that, fixed a theory T, any formula A, which is valid in any Boolean algebra making T true, is provable, i.e., there is a natural deduction derivation with no assumptions that has A as its conclusion.

In fact, we will prove a stronger result: in a theory T, for any finite set Γ of formulae and for any formula A, if $\bigwedge_{G \in \Gamma} [\![G]\!] \leq [\![A]\!]$ in any Boolean algebra which makes the theory T true, there is a natural deduction proof $\pi : \Gamma \vdash_T A$.

As a corollary, noticing that when $\Gamma = \emptyset$, $\bigwedge_{G \in \Gamma} [\![G]\!] = \top$, the previous result follows by anti-symmetry.

Preliminaries

The proof is complex and subtle.

In the first place, it is worth noticing that, if $\pi : \Gamma \vdash_T A$, then there is a finite $\Delta \subseteq \Gamma$ such that $\pi : \Delta \vdash_T A$. In fact, since any proof is a finite object, and any inference rule has a finite number of antecedents, only a finite number of assumptions may be used in a proof.

In this sense, the limit of having a finite Γ in the statement of the Completeness Theorem is not committing.

Strategy

Of course, the difficult aspect of the theorem lies in considering the totality of Boolean algebras.

The strategy behind the proof is

- construct a *canonical* Boolean algebra $\mathbb{B}$ which makes the axioms of T true, and which is 'easy' to manage;
- show that, for any other Boolean algebra $\mathbb{O}$, there is a function $e\colon \mathbb{B} \to \mathbb{O}$ which preserves the ordering relation;
- deduce that, up to isomorphisms, if $\bigwedge_{G\in\Gamma} [\![G]\!] \leq [\![A]\!]$ in every Boolean algebra, then $\bigwedge_{G\in\Gamma} [\![G]\!] \leq [\![A]\!]$ in $\mathbb{B}$, which is obvious, and vice versa, which is **not** obvious;
- prove that, for any finite set Γ of formulae and for any formula A, if $\bigwedge_{G\in\Gamma} [\![G]\!] \leq [\![A]\!]$ in $\mathbb{B}$, then there exists $\pi\colon \Gamma \vdash_T A$.

This strategy is general: many completeness results for most logical systems, follow this pattern. But there are exceptions...

Canonical model

The idea is to define a *canonical Boolean algebra*, in which truth and provability are the same notion.

Definition 6.1 (Canonical Boolean algebra)

Let T be a theory. Then the *canonical Boolean algebra* $\mathbb{B}(T)$ on T is the set $\{A\colon A \text{ is a formula in the language of } T\}/\sim$, where $A \sim B$ if and only if $A \vdash_T B$ and $B \vdash_T A$, together with the order defined by $[A]_\sim \leq_{\mathbb{B}(\mathbb{T})} [B]_\sim$ exactly when $A \vdash_T B$. For the sake of simplicity, when it is clear from the context, we omit the subscripts.

Notice how, posing A to be true exactly when $[\top]_\sim \leq [A]_\sim$, we get that $\varnothing \vdash A$ because $\top = \bigwedge \varnothing$.

But we have to show, first, that $\mathbb{B}(T)$ is a Boolean algebra.

An auxiliary result

Lemma 6.2

If $\pi : \Gamma \cup \{A\} \vdash_T B$ *and* $\theta : \Gamma \vdash_T A$*, then there is a proof* $\nu : \Gamma \vdash_T B$*.*

Proof. (i)

By induction on the structure of the proof π.

- if π is an instance of the assumption rule either $B \in \Gamma$, so ν coincides with π, which does not depend on A, or $B \equiv A$, thus $\nu = \theta$.
- if π is an instance of the axiom rule, $B \in T$, so $\nu = \pi$, which does not depend on A.
- if π is an instance of $\top$-introduction, $B \equiv \top$, so $\nu = \pi$, which does not depend on A.
- if π is an instance of $\bot$-elimination, by induction hypothesis, there is $\xi : \Gamma \vdash_T \bot$, so applying the $\bot$-elimination rule to ξ gives the required ν.

↪

An auxiliary result

↪ Proof. (ii)

- if π is an instance of the Law of Excluded Middle, $B \equiv C \vee \neg C$, so $\nu = \pi$, which does not depend on A.
- if π is an instance of $\wedge$-introduction, $B \equiv C \wedge D$, and, by induction hypothesis, there are $\xi \colon \Gamma \vdash_T C$ and $\mu \colon \Gamma \vdash_T D$, so the required ν is obtained by applying $\wedge$-introduction to ξ and μ.
- if π is an instance of $\wedge_1$-elimination, by induction hypothesis, there is $\xi \colon \Gamma \vdash_T B \wedge C$, so ν is obtained by applying $\wedge_1$-elimination to ξ.
- if π is an instance of $\wedge_2$-elimination, by induction hypothesis, there is $\xi \colon \Gamma \vdash_T C \wedge B$, so ν is obtained by applying $\wedge_2$-elimination to ξ.

↪

An auxiliary result

↪ Proof. (iii)

- if π is an instance of $\vee_1$-introduction, then $B \equiv C \vee D$ and, by induction hypothesis, there is $\xi : \Gamma \vdash_T C$, so ν is obtained by applying $\vee_1$-introduction to ξ.
- if π is an instance of $\vee_2$-introduction, then $B \equiv C \vee D$ and, by induction hypothesis, there is $\xi : \Gamma \vdash_T D$, so ν is obtained by applying $\vee_2$-introduction to ξ.
- if π is an instance of $\vee$-elimination, by induction hypothesis, there are $\xi : \Gamma \vdash_T C \vee D$, $\mu_C : \Gamma \cup \{C\} \vdash_T B$, and $\mu_D : \Gamma \cup \{D\} \vdash_T B$, so, applying $\vee$-elimination to ξ, μ_C, and μ_D the required ν is constructed.

↪

An auxiliary result

↪ Proof. (iv)

- if π is an instance of $\supset$-introduction, then $B \equiv C \supset D$ and, by induction hypothesis, there is $\xi : \Gamma \cup \{C\} \vdash_T D$, so ν is obtained by applying $\supset$-introduction to ξ.
- if π is an instance of $\supset$-elimination, by induction hypothesis, there are $\xi : \Gamma \vdash_T C \supset B$ and $\mu : \Gamma \vdash_T C$, so ν is constructed applying $\supset$-elimination to ξ and μ.
- if π is an instance of $\neg$-introduction, $B \equiv \neg C$ and, by induction hypothesis, there is $\xi : \Gamma \cup \{C\} \vdash_T \bot$, thus ν is obtained applying $\neg$-introduction to ξ.
- if π is an instance of $\neg$-elimination, by induction hypothesis there are $\xi : \Gamma \vdash_T \neg C$ and $\mu : \Gamma \vdash_T C$, so ν is constructed applying $\neg$-elimination to ξ and μ. □

Properties of the canonical model

Proposition 6.3

The relation $\sim$ is an equivalence relation.

Proof.

- By the assumption inference rule, $A \vdash_T A$, so $A \sim A$ for any formula A, i.e., $\sim$ is reflexive.
- If $A \sim B$, then $A \vdash_T B$ and $B \vdash_T A$, so $B \sim A$, too. That is, $\sim$ is symmetric.
- If $A \sim B$ and $B \sim C$ then there are $\pi_B \colon A \vdash_T B$ and $\pi_A \colon B \vdash_T A$, and $\theta_C \colon B \vdash_T C$ and $\theta_B \colon C \vdash_T B$. By Lemma 6.2, there are $\pi \colon A \vdash_T C$ and $\theta \colon C \vdash_T A$, that is, $A \sim C$, which means $\sim$ is transitive. □

Properties of the canonical model

Proposition 6.4

The relation $\leq_{\mathbb{B}(T)}$ is an ordering.

Proof.

- The relation $[A]_\sim \leq [B]_\sim$ does not depend on the choices of the representatives in the equivalence classes on $\sim$, in fact, if $[A]_\sim = [A']_\sim$ and $[B]_\sim = [B']_\sim$, then $A \sim A'$ and $B \sim B'$. So, by definition of $\sim$, $A' \vdash_T A$ and $B \vdash_T B'$. But, by definition of $\leq$, $A \vdash_T B$, thus, by Lemma 6.2 twice, $A' \vdash_T B'$, that is, $[A']_\sim \leq [B']_\sim$.
- By the assumption rule, $A \vdash_T A$, so $[A]_\sim \leq [A]_\sim$, i.e., $\leq$ is reflexive.
- If $[A]_\sim \leq [B]_\sim$ and $[B]_\sim \leq [C]_\sim$, then $A \vdash_T B$ and $B \vdash_T C$, so, by Lemma 6.2, $A \vdash_T C$, that is, $[A]_\sim \leq [C]_\sim$, i.e., $\leq$ is transitive.
- If $[A]_\sim \leq [B]_\sim$ and $[B]_\sim \leq [A]_\sim$, then $A \vdash_T B$ and $B \vdash_T A$, so, by definition of $\sim$, $A \sim B$, that is, $[A]_\sim = [B]_\sim$, i.e., $\leq$ is anti-symmetric. □

Properties of the canonical model

Proposition 6.5

$\mathbb{B}(T)$ *is a lattice.*

Proof.

- Consider $[A \wedge B]_\sim$: $[A \wedge B]_\sim \leq [A]_\sim$ since $A \wedge B \vdash_T A$ by $\wedge_1$-elimination; also, $[A \wedge B]_\sim \leq [B]_\sim$ since $A \wedge B \vdash_T B$ by $\wedge_2$-elimination. If $[C]_\sim \leq [A]_\sim$ and $[C]_\sim \leq [B]_\sim$, then $C \vdash_T A$ and $C \vdash_T B$, so $C \vdash_T A \wedge B$ by $\wedge$-introduction, thus $[C]_\sim \leq [A \wedge B]_\sim$. So, by definition of $\wedge$ in an order, $[A]_\sim \wedge [B]_\sim = [A \wedge B]_\sim$.
- Consider $[A \vee B]_\sim$: $[A]_\sim \leq [A \vee B]_\sim$ since $A \vdash_T A \vee B$ by $\vee_1$-introduction; also, $[B]_\sim \leq [A \vee B]_\sim$ since $B \vdash_T A \vee B$ by $\vee_2$-introduction. If $[A]_\sim \leq [C]_\sim$ and $[B]_\sim \leq [C]_\sim$, then $A \vdash_T C$ and $B \vdash_T C$, so $A \vee B \vdash_T C$ by $\vee$-elimination, thus $[A \vee B]_\sim \leq [C]_\sim$. So, by definition of $\vee$ in an order, $[A]_\sim \vee [B]_\sim = [A \vee B]_\sim$.

□

Properties of the canonical model

Proposition 6.6

$\mathbb{B}(T)$ *is a bounded lattice.*

Proof.

- For each formula A, $A \vdash_T \top$ by $\top$-introduction, so $[A]_\sim \leq [\top]_\sim$. Thus, by definition of $\top$ in a lattice, $\top = [\top]_\sim$.
- For each formula A, $\bot \vdash_T A$ by $\bot$-elimination, so $[\bot]_\sim \leq [A]_\sim$. Thus, by definition of $\bot$ in a lattice, $\bot = [\bot]_\sim$. □

Properties of the canonical model

Proposition 6.7

$\mathbb{B}(T)$ *is a distributive lattice.*

Proof. (i)

For any A, B, and C, $[A] \vee ([B] \wedge [C]) = [A] \vee [B \wedge C] = [A \vee (B \wedge C)]$ and $([A] \vee [B]) \wedge ([A] \vee [C]) = [A \vee B] \wedge [A \vee C] = [(A \vee B) \wedge (A \vee C)]$.

But $A \vee (B \wedge C) \vdash_T (A \vee B) \wedge (A \vee C)$ since

$$\dfrac{A \vee (B \wedge C) \quad \dfrac{\dfrac{[A]^*}{A \vee B} \vee \mathrm{I}_1 \quad \dfrac{[A]^*}{A \vee C} \vee \mathrm{I}_1}{(A \vee B) \wedge (A \vee C)} \wedge \mathrm{I} \quad \dfrac{\dfrac{\dfrac{[B \wedge C]^*}{B} \wedge \mathrm{E}_1}{A \vee B} \vee \mathrm{I}_2 \quad \dfrac{\dfrac{[B \wedge C]^*}{C} \wedge \mathrm{E}_2}{A \vee C} \vee \mathrm{I}_2}{(A \vee B) \wedge (A \vee C)} \wedge \mathrm{I}}{(A \vee B) \wedge (A \vee C)} \vee \mathrm{E}^*$$

↪

Properties of the canonical model

↪ Proof. (ii)

Also $(A \vee B) \wedge (A \vee C) \vdash_T A \vee (B \wedge C)$ since

$$\dfrac{\dfrac{(A \vee B) \wedge (A \vee C)}{A \vee B}\,\wedge E_1 \quad \dfrac{[A]^*}{A \vee (B \wedge C)}\,\vee I_1 \quad \begin{array}{c}[B]^* \\ \vdots \\ A \vee (B \wedge C)\end{array}}{A \vee (B \wedge C)}\,\vee E^*$$

where the third antecedent is

$$\dfrac{\dfrac{(A \vee B) \wedge (A \vee C)}{A \vee C}\,\wedge E_2 \quad \dfrac{[A]^\dagger}{A \vee (B \wedge C)}\,\vee I_1 \quad \dfrac{\dfrac{B \quad [C]^\dagger}{B \wedge C}\,\wedge I}{A \vee (B \wedge C)}\,\vee I_2}{A \vee (B \wedge C)}\,\vee E^\dagger$$

Thus, $(A \vee B) \wedge (A \vee C) \sim A \vee (B \wedge C)$, and the conclusion follows. □

Properties of the canonical model

Proposition 6.8

$\mathbb{B}(T)$ *is a complemented lattice.*

Proof.

Consider, for any formula A, $[\neg A]$: $[A] \wedge [\neg A] = [A \wedge \neg A] = [\bot] = \bot$, since $\bot \vdash_T A \wedge \neg A$ by $\bot$-elimination, and

$$\dfrac{\dfrac{A \wedge \neg A}{A}\,\wedge\mathsf{E}_1 \qquad \dfrac{A \wedge \neg A}{\neg A}\,\wedge\mathsf{E}_2}{\bot}\,\neg\mathsf{E}$$

Also, $[A] \vee [\neg A] = [A \vee \neg A] = [\top] = \top$, since $A \vee \neg A \vdash_T \top$ by $\top$-introduction, and $\top \vdash_T A \vee \neg A$ by the Law of Excluded Middle. □

Corollary 6.9

$\mathbb{B}(T)$ *is a Boolean algebra.*

Classifying models

Proposition 6.10

Fixed a theory T, let $\mathbb{O}$ be any Boolean algebra and let $\nu\colon V \to \mathbb{O}$ be any assignment of variables on it such that $[\![A]\!] = \top$ for any $A \in T$. If $[B]_\sim \leq_{\mathbb{B}(T)} [C]_\sim$, then $[\![B]\!]_{\mathbb{O}} \leq_{\mathbb{O}} [\![C]\!]_{\mathbb{O}}$.

Proof.

If $[B]_\sim \leq_{\mathbb{B}(T)} [C]_\sim$, then there is $\pi\colon B \vdash_T C$ by definition of $\leq_{\mathbb{B}(T)}$. Thus, by the proof of the Soundness Theorem 5.21, applied in the $\mathbb{O}$ Boolean algebra with the ν assignment, $[\![B]\!]_{\mathbb{O}} \leq_{\mathbb{O}} [\![C]\!]_{\mathbb{O}}$. □

Classifying models

Definition 6.11 (Canonical map)

Fixed a theory T, let $\mathbb{O}$ be any Boolean algebra and let $\nu\colon V \to \mathbb{O}$ be any assignment of variables on it such that $[\![A]\!] = \top$ for any $A \in T$. Then, the map $\xi_{\mathbb{O}}\colon \mathbb{B} \to \mathbb{O}$, defined by $[B]_{\sim} \mapsto [\![B]\!]_{\mathbb{O}}$, is the *canonical map* to $\mathbb{O}$.

This definition does not depend on the choice of the representatives in $\mathbb{B}$. In fact, if $[A] = [A']$, then, $[A] \leq [A']$ and $[A] \leq [A']$, so, by Proposition 6.10, $[\![A]\!] \leq [\![A']\!]$ and $[\![A']\!] \leq [\![A]\!]$ in $\mathbb{O}$, thus, by anti-symmetry, $[\![A]\!] = [\![A']\!]$.

Moreover, the canonical map, preserves the ordering of $\mathbb{B}$.

Completeness

Theorem 6.12 (Completeness)

Fixed a theory T, for any finite set Γ of formulae and for any formula A, if $\bigwedge_{G\in\Gamma}[\![G]\!] \leq [\![A]\!]$ in any Boolean algebra and any assignment of variables which makes the theory T true, then there is a natural deduction proof $\pi : \Gamma \vdash_T A$.

Proof.

If $\bigwedge_{G\in\Gamma}[\![G]\!] \leq [\![A]\!]$, then $[\![\bigwedge_{G\in\Gamma} G]\!] \leq [\![A]\!]$.
Since this fact holds in any Boolean algebra, it holds also in $\mathbb{B}(T)$, the canonical Boolean algebra on T. And, because of the way interpretation is defined in $\mathbb{B}(T)$, $[\bigwedge_{G\in\Gamma} G] \leq [A]$.
So, by definition of $\leq$ in $\mathbb{B}(T)$, there is $\pi : \bigwedge_{G\in\Gamma} G \vdash_T A$. Noticing that $\Gamma \vdash_T \bigwedge_{G\in\Gamma} G$ by iterating the $\wedge$-introduction rule, by Proposition 6.2 it follows $\Gamma \vdash_T A$. □

Completeness

Corollary 6.13

If $[\![A]\!] = \top$ *in every Boolean algebra and with any assignment of variables making the theory* T *true, then there is a proof* $\pi : \vdash_T A$.

Proof.

If $[\![A]\!] = \top$, then $\top = [\![\top]\!] \leq [\![A]\!]$, being $\leq$ reflexive. By the Completeness Theorem, the result follows immediately. □

Classifying models

In fact, we have another result for free: any *model* for a theory T, i.e., any Boolean algebra $\mathbb{O}$ together with an assignment of variables, is described by its canonical map $\xi_{\mathbb{O}}$.

In a sense, all the models of a theory T can be synthesised from the canonical model applying a canonical map. It is tempting to identify the models with the class of canonical maps...

...but this is another story which leads very far. And we will not pursue it during this course.

References

The proof has been adapted from the one in topos theory, which is illustrated in Section D of *Peter Johnstone*, Sketches of an Elephant: A Topos Theory Compendium, Oxford Logic Guides 43, Oxford University Press, (2003), ISBN 978-0198524960.

The notion of classifying model is central in the topos-theoretic approach, and, in some way, it goes back to Grothendieck's work. Again, Johnstone's book is a good starting point.

Mathematical Logic

Lecture 7

Dr Marco Benini

marco.benini@uninsubria.it

Dipartimento di Scienza e Alta Tecnologia
Università degli Studi dell'Insubria

a.a. 2016/17

Syllabus

First order logic:

- Language
- Substitution
- Natural deduction
- Examples

First-order logic

Propositional logic is a toy system. A very useful one, indeed, but, still, it has not enough expressive power to allow us to describe any useful mathematical theory, e.g., arithmetic or set theory.

Although propositional theories are very well-behaved, as we have seen, we want to use logic as a tool to do real mathematics. And, to achieve this objective, we need to speak about objects.

The main novelty in first-order logic is that the language is able to identify objects, and to write formulae on them. As already said, we allow quantification to freely range over objects, but not over sets of objects, or other collections/structures of objects.

Although outside the scope of the present course, higher-order logics, which allow extended quantification, cannot be complete. And first-order logic is, in a way, at the borderline for completeness, as we will illustrate in due time.

Language

Definition 7.1 (Signature)

A *signature* $\Sigma = \langle S; F; R \rangle$ is composed by

- a set S of symbols for *sorts*.
- a set F of symbols for *functions*. Each symbol $f \in F$ is uniquely associated with a *type* $s_1 \times \cdots \times s_n \to s_0$, with $s_i \in S$ for each $0 \leq i \leq n$. When $n = 0$, we say that f is a *constant* of type s_0.
- a set R of symbols for *relations*. Each symbol $r \in R$ is uniquely associated with a *type* $s_1 \times \cdots \times s_n$, with $s_i \in S$ for each $1 \leq i \leq n$. When $n = 0$, we say that r is a *propositional constant*.

The notation $f : s_1 \times \cdots \times s_n \to s_0 \in F$ and $r : s_1 \times \cdots \times s_n \in R$ means that f is a function symbol whose type is $s_1 \times \cdots \times s_n \to s_0$, and r is a relation symbol whose type is $s_1 \times \cdots \times s_n$, respectively. Also, we require that S, F, and R do not contain the logical connectives and quantifiers.

A signature describes a first-order language: sorts stands for collection of elements, functions are used to denote elements, while relations are used to form basic formulae.

Example 7.2

The signature

$$\mathscr{N} = \langle \{\mathbb{N}\}; \{0 : \mathbb{N}, S : \mathbb{N} \to \mathbb{N}; + : \mathbb{N} \times \mathbb{N} \to \mathbb{N}, \cdot : \mathbb{N} \times \mathbb{N} \to \mathbb{N}\}; \{= : \mathbb{N} \times \mathbb{N}\} \rangle$$

specifies the basic language for arithmetic. There is one sort, which, in the intended interpretation, stands for the collection of natural numbers. There is constant, 0, denoting the zero natural number, there is a function S, which stands for 'successor', denoting the next natural number, so that, in the intended interpretation, $S(5) = 6$, while the functions $+$ and $\cdot$ denote addition and multiplication.
There is only one relation symbol, denoting equality.
Of course, the theory of arithmetic should be devised in such a way that, as far as possible, the formal behaviour, that is, what we can prove, conforms to the intended interpretation.

Language

Example 7.3

The signature $\mathscr{G} = \langle \{G\}; \{1\colon G, \cdot\colon G \times G \to G, \_^{-1}\colon G \to G\}; \{=\colon G \times G\}\rangle$ describes the language of the theory of groups.

Example 7.4

The signature $\mathscr{O} = \langle \{O\}; \emptyset; \{\leq\colon O \times O\}\rangle$ describes the language of the theory of orders.

Example 7.5

The signature $\mathscr{L} = \langle \{E, L\}; \{\mathsf{nil}\colon L, \mathsf{cons}\colon E \times L \to L\}; \{=_E\colon E \times E, =_L\colon L \times L\}\rangle$ defines the language of the theory of lists. A computer scientist would say it defines the *data type* of lists.

Terms

The first-order language has two-purposes: to provide a syntax to denote elements in the universe, i.e., in the collections denoted by the sorts, and to provide a syntax to denote properties of those elements.

The first issue is addressed by *terms*.

Definition 7.6 (Term)

Let $\Sigma = \langle S; F; R \rangle$ be a signature, and let V be an infinite set of symbols, called *variables*, such that $V \cap (S \cup F \cup R) = \emptyset$. Also, assume that each variable $x \in V$ has a uniquely associated type $s \in S$, denoted by $x : s$. We require that there is an infinite amount of variables for each type $s \in S$. A *term*, along with the set of its *free variables*, is inductively defined as:

- if $x : s \in V$, then x is a term of type s, and $\mathrm{FV}(x) = \{x\}$;
- if $f : s_1 \times \cdots \times s_n \to s_0 \in F$ and $t_1, \ldots, t_n$ are terms of type $s_1, \ldots, s_n$, respectively, then $f(t_1, \ldots, t_n)$ is a term of type s_0, and $\mathrm{FV}(f(t_1, \ldots, t_n)) = \bigcup_{i=1}^{n} \mathrm{FV}(t_i)$.

We use the notation $t : s$ to say that the term t has type s.

Terms

Example 7.7

Using the signature $\mathcal{N}$ of arithmetic, 0, $S(0)$, $S(S(0))$, ... are terms of type $\mathbb{N}$. Also $+(x,0)$ and $\cdot(x,+(S(0),S(S(0))))$ are terms of type $\mathbb{N}$. Notice how $x+0$ and $x(1+2)$ are **not** terms.

To cope with the problem of expressing the standard notation of mathematics within the rigid syntax of logical terms, we will formally introduce definitions later.

Formulae

As terms are used to denote elements, formulae are used to denote properties of elements. The syntax is similar to propositional logic, with two important differences: we have atomic formulae instead of propositional variables, and we have quantifiers.

Definition 7.8 (Formula)

Fixed a signature $\Sigma = \langle S;\ F; R\rangle$ and a set of variables as for terms, a *formula*, along with the set of its *free variables*, is inductively defined as

- $\top$ and $\bot$ are formulae, and $\mathrm{FV}(\top) = \mathrm{FV}(\bot) = \emptyset$.
- if $r : s_1 \times \cdots \times s_n \in R$ is a relation symbol, and $t_1 : s_1, \ldots, t_n : s_n$ are terms, then $r(t_1, \ldots, t_n)$ is an *atomic* formula, and $\mathrm{FV}\left(r(t_1, \ldots, t_n)\right) = \bigcup_{i=1}^{n} \mathrm{FV}(t_i)$.
- if A and B are formulae, so are $\neg A$, $A \wedge B$, $A \vee B$, and $A \supset B$, and $\mathrm{FV}(\neg A) = \mathrm{FV}(A)$, $\mathrm{FV}(A \wedge B) = \mathrm{FV}(A \vee B) = \mathrm{FV}(A \supset B) = \mathrm{FV}(A) \cup \mathrm{FV}(B)$.
- if $x : s$ is a variable and A is a formula, so are $\forall x : s.\, A$ and $\exists x : s.\, A$, and $\mathrm{FV}(\forall x : s.\, A) = \mathrm{FV}(\exists x : s.\, A) = \mathrm{FV}(A) \setminus \{x\}$.

Formulae

There are two main differences between first-order formulae and propositional ones:

- instead of propositional variables, we have atomic formulae, which link the formulae with terms by means of a relation;
- there are quantified formulae, where the variable is **not** free. We say that quantified variables are *bounded*.

The notion of bounded variable is not new: for example, the expression $\int_a^b f(x)\,\mathrm{d}x$ does not really depend on the variable x. In fact, the x is a placeholder, to give some name to the argument of the f function. A bounded variable does not denote a value, but rather it acts as a placeholder which allows to write a formula or a term. Its meaning is controlled by the quantifier, and not by the way variables are interpreted, as in the integral, the x does not denote a real or complex number, but rather what is allowed to vary in the function.

Substitution

Variables are subject to a fundamental operation: substitution. In fact, from a formula A where the variable x appears free, we may obtain another formula, $A[t/x]$, where the term t is substituted for x. For example, in the language of arithmetic, x can be substituted in $x+0=x$ to obtain $2+0=2$.

Substitution is fundamental in describing the inference rules governing quantifiers. And bounded variables make substitution not immediately intuitive.

There are many equivalent ways to describe the substitution operation: we will use a method which is not the most immediate, but it will become very handy later in the course.

Substitution

Definition 7.9 (Substitution on terms)

Fixed a signature and a term t on it, the *substitution* of the variable $x : s$ with the term $r : s$, yielding $t[r/x]$, is defined by induction on the structure of the term t:

- if $t \equiv x$, then $t[r/x] = r$;
- if t is a variable, but $t \not\equiv x$, $t[r/x] = t$;
- if $t \equiv f(t_1, \ldots, t_n)$, then $t[r/x] = f(t_1[r/x], \ldots, t_n[r/x])$.

Notice that the substitution operation is defined only when t and x share the same type.

Substitution

Definition 7.10 (Substitution on formulae)

Fixed a signature and a formula A on it, the *substitution* of the variable $x : s$ with the term $t : s$, yielding $A[t/x]$, is defined by induction on the structure of the formula A:

- if $A \equiv \top$ or $A \equiv \bot$, then $A[t/x] = A$;
- if $A \equiv r(t_1, \ldots, t_n)$, then $A[t/x] = r(t_1[t/x], \ldots, t_n[t/x])$;
- if $A \equiv \neg B$, then $A[t/x] = \neg B[t/x]$;
- if $A \equiv B \wedge C$, $A \equiv B \vee C$, or $A \equiv B \supset C$, then $A[t/x] = B[t/x] \wedge C[t/x]$, $A[t/x] = B[t/x] \vee C[t/x]$, or $A[t/x] = B[t/x] \supset C[t/x]$, respectively;
- if $A \equiv \forall y : r.B$, or $A \equiv \exists y : r.B$, and $y : r \equiv x : s$, then $A[t/x] = A$;
- if $A \equiv \forall y : r.B$, or $A \equiv \exists y : r.B$, and $y : r \not\equiv x : s$, then $A[t/x] = \forall z : r.(B[z/y])[t/x]$, or $A[t/x] = \exists z : r.(B[z/y])[t/x]$, respectively, where $z : r \notin \mathrm{FV}(B) \cup \mathrm{FV}(t)$.

Substitution

The first clauses in the definition are obvious: we substitute the variable x with the term t where it appears.

The last but one clause means that a bounded variable cannot be substituted: this is simple to understand, as it does not make sense to substitute x with 5 in the formula $\exists x : \mathbb{N}. x^2 = x^3$. In fact, the formula is true, because $1^2 = 1 = 1^3$, but, evidently, it happens just for **some** values of x, which the existential quantifier is meant to single out.

The last clause is a bit cryptic. It says that, before performing the substitution of x with t on the quantified formula B, we should rename the quantified variable y with a **new** variable, which does not appear in B and t.

An example may clarify why this must be done: let $A \equiv \exists x : \mathbb{N}. x + y = 2y$, and let $t \equiv 2x$. If we do not rename variables, $A[t/y]$ would give $\exists x : \mathbb{N}. x + 2x = 2(2x)$, that is, $\exists x : \mathbb{N}. 3x = 4x$. We notice the A holds whenever $x = y$, but $A[t/y]$ does not. The problem is that the x in t and the one in A should be kept distinct—and we do this by renaming before performing the substitution.

Definitions

The language of first-order logic is cumbersome. Despite the fact that we already use a simplified notation, avoiding unneeded parentheses and hiding what can be immediately inferred from the context, the formal nature of the language is far distant from the reality of the mathematical practice.

On the contrary, the formal nature of the language is what allows it to be analysed: we constantly use induction on the structure of the language (terms, formulae, proofs) as our main proving instrument.

There is a way in between: we can construct a reasonable formal language by taking a basic formal language, and enriching it with *syntactical sugar*. This does not change the formal nature of the language, but allows to make the language much closer to the standard practice.

This practise takes place by allowing syntactical construction which are not part of the formal language, but, still, can be directly translated into the formal language. This construction is called *definition*, and it has to follow a few, precise rules.

Definitions

Definition 7.11 (Function definition)

Fixed a first-order language with equality, let f be a new symbol. Whenever it holds that $\forall x_1 : s_1 \ldots \forall x_n : s_n . \exists y : s_0 . F \wedge \forall z : s_0 . F[z/y] \supset z = y$, with $\mathrm{FV}(F) \subseteq \{x_1, \ldots, x_n, y\}$, then $f : s_1 \times \cdots \times s_n \to s_0$ can be used as an additional function symbol, since it can be removed from the language by the rule

$$A[f(t_1, \ldots, t_n)/z] = \exists z : s_0 . A \wedge (F[z/y])[t_1/x_1, \ldots, t_n/x_n] \wedge \\ \wedge \forall w : s_0 . (F[w/y])[t_1/x_1, \ldots, t_n/x_n] \supset z = w$$

for any formula A, and with the obvious extensions to the definition of substitution. As far as a different syntax is non-ambiguous, we allow it in place of the standard functional syntax.

Definitions

Definition 7.12 (Relation definition)

Fixed a first-order language, let r be a new symbol. Then $r : s_1 \times \cdots \times s_n$ can be used as an additional relation symbol standing for the formula R whenever $\mathrm{FV}(R) = \{x_1, \ldots, x_n\}$, since it can be removed by substituting $R(t_1, \ldots, t_n)$ whenever $r(t_1, \ldots, t_n)$ occurs in any formula A. Again, as far as the syntax is non-ambiguous, we allow fancy syntactical constructions.

Notice that there is no way to define new sorts. This happens because defining new sorts require sophisticated rules which cannot be easily managed by translating into the original language.

Definitions

Example 7.13

Consider the language generated by the signature:

$$\langle \{\mathbb{N}\}; \{0 : \mathbb{N}, \mathrm{succ} : \mathbb{N} \to \mathbb{N}, \mathrm{add} : \mathbb{N} \times \mathbb{N} \to \mathbb{N}, \mathrm{times} : \mathbb{N} \times \mathbb{N} \to \mathbb{N}\}; \{\mathrm{eq} : \mathbb{N} \times \mathbb{N}\} \rangle$$

Then, the syntax $x + y$ stands for $\mathrm{add}(x, y)$, xy stands for $\mathrm{times}(x, y)$, and $x = y$ stands for $\mathrm{eq}(x, y)$. The last definition is a relation symbol definition, while the first two definitions are function symbol definitions, corresponding to the formulae

$$\forall x : \mathbb{N}. \forall y : \mathbb{N}. \exists z : \mathbb{N}. \mathrm{eq}(\mathrm{add}(x, y), z) \wedge \forall w : \mathbb{N}. \mathrm{eq}(\mathrm{add}(x, y), w) \supset \mathrm{eq}(z, w)$$

and

$$\forall x : \mathbb{N}. \forall y : \mathbb{N}. \exists z : \mathbb{N}. \mathrm{eq}(\mathrm{times}(x, y), z) \wedge \forall w : \mathbb{N}. \mathrm{eq}(\mathrm{times}(x, y), w) \supset \mathrm{eq}(z, w)$$

that we must prove.

Definitions

Example 7.14

Consider any first-order language with equality. Then we may add a new family of relation symbols $\exists! x : s. A$ with $x : s$ a variable and A a formula, which stands for $\exists x : s. A \wedge \forall z : s. A[z/x] \supset z = x$, with $z : s \notin \mathrm{FV}(A)$. Syntactically, this appears as a new form of quantification, which is read as 'uniquely exists'.

Natural deduction

Fixed any first-order language, the definition of *theory* follows the one already given in the propositional case.

The same holds for the definition of *proof* and the other related terms, except that the collection of inference rules contains four new rules, to deal with quantifiers. They are illustrated in the next slides.

When the language contains equality, we require the presence of other inference rules, detailed in the next slides.

The modular composition of inference rules in natural deduction explains why we chose this deduction system instead of one of the many others in literature: all the deduction systems in this course are obtained by adding or deleting a few rules from the propositional or the first-order case.

Natural deduction

Following the previous notation, the rules for universal quantification are

$$\frac{A}{\forall x : s.A}\,\forall\mathrm{I} \qquad \frac{\forall x : s.A}{A[t/x]}\,\forall\mathrm{E}$$

provided that

- in $\forall E$, t is a term of type s;
- in $\forall I$, the variable $x : s$ does not *occur free in the proof* of the antecedent, which means that, for every assumption G, $x : s \notin \mathrm{FV}(G)$. This condition is, sometimes, referred to by saying that $x : s$ is an *eigenvariable*.

Notice the similarity between the rules for $\forall$ and for $\wedge$.

Natural deduction

Similarly, the rules for existential quantification are

$$\frac{A[t/x]}{\exists x : s.\, A}\,\exists\mathsf{I} \qquad \frac{\exists x : s.\, B \quad \begin{array}{c}[B]\\ \vdots \\ A\end{array}}{A}\,\exists\mathsf{E}$$

provided that

- in $\exists I$, t is a term of type s;
- in $\exists E$, the variable $x : s$ does not occur free in the proof of the second antecedent, that is, for every assumption G in the second subproof, except for B, $x : s \notin \mathrm{FV}(G)$. Again, $x : s$ is said to be an eigenvariable. Notice how this inference rule discharges the assumption B.

Notice the similarity between the rules for $\exists$ and for $\vee$.

Natural deduction

Equality is a special relation, and this is captured in a series of ad-hoc inference rules. When the language has an equality relation for some sort s, it is subject to the following rules:

$$\frac{}{\forall x : s.\, x = x}\ \text{refl} \qquad \frac{}{\forall x : s.\forall y : s.\, x = y \supset y = x}\ \text{sym}$$

$$\frac{}{\forall x : s.\forall y : s.\forall z : s.\, x = y \wedge y = z \supset x = z}\ \text{trans}$$

$$\frac{A[t/x] \quad t = r}{A[r/x]}\ \text{subst}$$

$$\frac{}{\forall x_1 : s_1 \ldots \forall x_n : s_n.\exists! z : s_0.\, z = f(x_1, \ldots, x_n)}\ \text{fun}$$

where, t and r are terms of type s, and $f : s_1 \times \cdots \times s_n \to s_0$ is a function symbol of the language.

Examples

Example 7.15

$$\dfrac{\dfrac{\dfrac{\dfrac{[P]^1}{\exists x : s.\, P}\,\exists\mathsf{I} \qquad [\neg\exists x : s.\, P]^2}{\bot}\,\neg\mathsf{E}}{\neg P}\,\neg\mathsf{I}^1}{\forall x : s.\, P}\,\forall\mathsf{I}}{(\neg\exists x : s.\, P) \supset \forall x : s.\, \neg P}\,\supset\mathsf{I}^2$$

By applying the double-negation law ($A = \neg\neg A$), and taking $P \equiv \neg A$, we get that $(\neg\exists x : s.\, \neg A) \supset \forall x : s.\, A$.

Examples

Example 7.16

$$\dfrac{\dfrac{\dfrac{[\exists x : s.\, P]^1 \qquad \dfrac{[P]^2 \qquad \dfrac{[\forall x : s.\, \neg P]^3}{\neg P}\forall E}{\bot}\neg E}{\bot}\exists E^2}{\neg\exists x : s.\, P}\neg I^1}{(\forall x : s.\, \neg P) \supset \neg\exists x : s.\, P}\supset I^3$$

Putting $P \equiv \neg A$ and applying the double negation law, one gets that $\forall x : s.\, A = \neg\exists x : s.\, \neg A$.

Examples

Example 7.17

$$\dfrac{\dfrac{\dfrac{[\exists x : s. \neg P]^1 \qquad \dfrac{\dfrac{[\forall x : s. P]^2}{P}\,\forall \mathsf{E} \qquad \dfrac{[\neg P]^3}{}}{\bot}\,\neg \mathsf{E}}{\bot}\,\exists \mathsf{E}^3}{\neg \forall x : s. P}\,\neg \mathsf{I}^2}{(\exists x : s. \neg P) \supset \neg \forall x : s. P}\,\supset \mathsf{I}^1$$

Examples

Example 7.18

$$\dfrac{\dfrac{}{(\exists x : s.\, \neg P) \vee \neg(\exists x : s.\, \neg P)}\,\text{lem} \qquad \dfrac{\dfrac{[\neg\exists x : s.\, \neg P]^1 \qquad \dfrac{\dfrac{\forall x : s.\, P \qquad [\neg\forall x : s.\, P]^2}{\bot}\,\neg\text{E}}{\exists x : s.\, \neg P}\,\bot\text{E}}{\exists x : s.\, \neg P}\,\vee\text{E}^1}{(\neg\forall x : s.\, P) \supset \exists x : s.\, \neg P}\,\supset\text{I}^2$$

Examples

Example 7.19

To show that the restrictions on variables in the introduction rule of the universal quantifier is essential, consider the following counterexample. Let $x : s \in \mathrm{FV}(P)$.

$$\dfrac{\dfrac{\dfrac{[P]^1}{\forall x : s.\, P}\,\forall \mathrm{I}}{P \supset \forall x : s.\, P}\,\supset \mathrm{I}^1}{\forall x : s.\, (P \supset \forall x : s.\, P)}\,\forall \mathrm{I}$$

The instance of the $\forall I$ rule on the top is invalid, since $x : s$ appear in the assumptions which are undischarged in that moment of the proof.
In arithmetic, if P stands for 'x is even', the conclusion allows to prove that, since $P[0/x]$ is true, every natural number is even!

Examples

Example 7.20

Another counterexample, showing why the restriction on variables is essential in the elimination rule for the existential quantifier, is the following. Again, let $x : s \in \mathrm{FV}(P)$.

$$\dfrac{\dfrac{\dfrac{\dfrac{[\exists x : s.\, P]^1 \qquad \dfrac{[P \supset Q]^2 \quad [P]^3}{Q}\supset\mathrm{E}}{Q}\exists\mathrm{E}^3}{(\exists x : s.\, P) \supset Q}\supset\mathrm{I}^1}{(P \supset Q) \supset ((\exists x : s.\, P) \supset Q)}\supset\mathrm{I}^2}{\forall x : s.((P \supset Q) \supset ((\exists x : s.\, P) \supset Q))}\forall\mathrm{I}$$

Inside arithmetic, let $Q \equiv \bot$, so the conclusion reduces to $\forall x : s.(\neg P \supset \neg \exists x : s.\, P)$. If P stands for 'x is even', since $P[1/x]$ is false, the conclusion allows to deduce that there is no even natural number!

References

Usually, first-order logic is presented in a simplified way, by avoiding the multi-sorted language, and by using a reduced number of connectives. Although this approach simplifies the initial presentation, it makes difficult to pass to other logical system, e.g., intuitionistic logic, and to deal with real mathematical theories, where multiple sorts are often present.

A good text which introduces the first-order language in a formal way is *John Bell* and *Moshé Machover*, A Course in Mathematical Logic, North-Holland, (1977), ISBN 0-7204-28440, which covers our treatment of definitions, too.

Natural deduction is described in many textbooks. This lesson follows *A.S. Troelstra* and *H. Schwichtenberg*, Basic Proof Theory, Cambridge Tracts in Theoretical Computer Science 43, Cambridge: Cambridge University Press, (1996). The counterexamples have been taken from that text.

Mathematical Logic

Lecture 8

Dr Marco Benini

marco.benini@uninsubria.it

Dipartimento di Scienza e Alta Tecnologia
Università degli Studi dell'Insubria

a.a. 2016/17

Syllabus

First order logic:

- Informal meaning
- Semantics
- Examples
- Soundness

Informal meaning

Fixed a signature $\langle S; F; R \rangle$, the intended interpretation of a sort $s \in S$ is a specific set; the intended interpretation of a function symbol is a function; and the intended interpretation of a relation symbol is a relation.

The intended meaning of equality, $=: s \times s$, when present in the language, is the identity of its arguments.

Thus, the intended meaning of a term is an element, which is identified via the interpretation of functions and the evaluation of variables, in the universe, the collection of all the sets denoted by sorts.

Informal meaning

In turn, formulae stands for a truth value, either true or false, as in the propositional case. And connectives have the intended propositional meaning, we already illustrated.

Atomic formulae, $r(t_1, \ldots, t_n)$, are true when the argument $(t_1, \ldots, t_n)$ is in the relation denoted by r.

A formula is universally valid, that is, $\forall x : s. A$ holds, when A is true in whatever way we interpret x as an element of the set denoted by s.

Symmetrically, a formula is existentially valid, that is, $\exists x : s. A$ holds, when there is an element e in the set denoted by s such that interpreting x as e makes A true.

Semantics

The standard semantics for first-order logic, due to Alfred Tarski, directly formalises the intended interpretation.

Definition 8.1 (Σ-structure)

Let $\Sigma = \langle S; F, R \rangle$ be a first-order signature.
Then, a Σ-*structure* $\mathscr{M} = \langle U; \mathscr{F}; \mathscr{R} \rangle$ is composed by

- a collection $U = \{u_s\}_{s \in S}$ of non-empty sets, called the *universe*,
- a collection of functions over the universe
  $\mathscr{F} = \left\{ g_f : u_{s_1} \times \cdots \times u_{s_n} \to u_{s_0} \mid f : s_1 \times \cdots \times s_n \to s_0 \in F \right\}$,
- a collection of relations over the universe
  $\mathscr{R} = \left\{ \rho_r : u_{s_1} \times \cdots \times u_{s_n} \mid r : s_1 \times \cdots \times s_n \in R \right\}$.

Semantics

To make clear the relation between a signature and a Σ-structure, we use the following notation:

- for each $s \in S$, $[\![s]\!] = u_s$;
- for each $f : s_1 \times \cdots \times s_n \to s_0 \in F$, $[\![f]\!] = g_f$;
- for each $r : s_1 \times \cdots \times s_n \in R$, $[\![r]\!] = \rho_r$.

This is called the *interpretation of the signature* in the Σ-structure.

Semantics

Definition 8.2 (Interpretation of terms)

Let $\Sigma = \langle S; F, R \rangle$ be a signature, and let $\mathscr{M}$ be a Σ-structure, with the notation as before. Also, let $\nu = \{\nu_s\}_{s \in S}$ be a collection of functions $\nu_s \colon \{v \colon v \colon s \in V\} \to [\![s]\!]$, mapping the variables of type s into the corresponding set $[\![s]\!]$.
Then, a term t is interpreted according to the following inductive definition on its structure:

- if $t \in V$ is a variable of type s, then $[\![t]\!] = \nu_s(t)$;
- if $t \equiv f(t_1, \ldots, t_n)$, then $[\![t]\!] = [\![f]\!]([\![t_1]\!], \ldots, [\![t_n]\!])$.

Semantics

Definition 8.3 (Interpretation of formulae)

Let $\Sigma = \langle S; F, R\rangle$ be a signature, let $\mathscr{M}$ be a Σ-structure, and let ν be an *evaluation of variables*, with the notation as before.
Then, a formula A is interpreted according to the following inductive definition on its structure:

- if $A \equiv \top$, $[\![A]\!] = 1$;
- if $A \equiv \bot$, $[\![A]\!] = 0$;
- if $A \equiv r(t_1, \ldots, t_n)$, $[\![A]\!] = 1$ if $([\![t_1]\!], \ldots, [\![t_n]\!]) \in [\![r]\!]$, and $[\![A]\!] = 0$ otherwise;
- if $A \equiv \neg B$, $A \equiv B \wedge C$, $A \equiv B \vee C$, $A \equiv B \supset C$, then $[\![A]\!]$ is defined as in the truth-table semantics;
- if $A \equiv \forall x\colon s.\, B$ or $A \equiv \exists x\colon s.\, B$, let $\xi = \{\xi_s\}_{s \in S}$ be an evaluation of variables such that, $\xi_\alpha = \nu_\alpha$, for each $\alpha \neq s$, and $\xi_s(v) = \nu_s(v)$ for each $v \neq x$. Then, $[\![\forall x\colon s.\, B]\!] = 1$ if, for all the possible ξ, $[\![B]\!] = 1$, and $[\![\forall x\colon s.\, B]\!] = 0$ otherwise. Also, $[\![\exists x\colon s.\, B]\!] = 1$ if, there is a ξ such that $[\![B]\!] = 1$, and $[\![\exists x\colon s.\, B]\!] = 0$ otherwise.

Semantics

We stipulate that, when equality is in the language, $[\![t_1 = t_2]\!] = 1$ exactly when $[\![t_1]\!] = [\![t_2]\!]$.

If one prefers, $[\![=_s]\!]$, the equality on the sort s, represents the *diagonal relation* $\{(x,x) : x \in [\![s]\!]\}$.

It is worth remarking that equality is always typed: $t_1 = t_2$ is a valid formula if and only if t_1 and t_2 are terms of the same sort s, and the relation $=$ should be read as a shorthand for $=_s$, which stands for the diagonal relation on the set denoted by the sort s.

Examples

Example 8.4

Fix the signature of arithmetic, and consider the standard model of natural numbers. Then, the formula $S0 + S0 = SS0$ is interpreted in $[\![S0 + S0 = SS0]\!] = 1$ since

1. $[\![S0 + S0]\!] = [\![+]\!]([\![S0]\!], [\![S0]\!]) = +([\![S]\!]([\![0]\!]), [\![S]\!]([\![0]\!])) = +(1 + 0, 1 + 0) = 1 + 1 = 2$;
2. $[\![SS0]\!] = [\![S]\!]([\![S0]\!]) = [\![S]\!]([\![S]\!]([\![0]\!])) = 1 + (1 + 0) = 1 + 1 = 2$;
3. $[\![S0 + S0 = SS0]\!] = 1$ if and only if $[\![S0 + S0]\!] = [\![SS0]\!]$, that is, if and only if $2 = 2$.

Examples

Example 8.5

Fix the signature of arithmetic, and consider the standard model of natural numbers. Let consider $[\![x = (SS0)y]\!]$. Applying the definition of semantics, $[\![x = (SS0)y]\!] = 1$ if and only if $[\![x]\!] = 2[\![y]\!]$, that is, if and only if x is interpreted in a number which is two times the value y is interpreted in.

So, if x is interpreted in 6 and y in 3, the formula is true, while if x is interpreted in 6, but y in 5, the formula is false.

Examples

Example 8.6

Fix the signature of arithmetic, and consider the standard model of natural numbers. Consider $[\![\exists x.x = (SS0)x]\!]$. Applying the definition of semantics, $[\![\exists x.x = (SS0)x]\!] = 1$ if and only if there is an assignment ξ of variables, identical to the one fixed in the model except for the value it assigns to x, such that $[\![x = (SS0)x]\!] = 1$. But, whenever $\xi(x) = 0$, $[\![x = (SS0)x]\!] = 1$ since both sides evaluate to 0, so the initial formula is true.

Consider $[\![\forall x.x = (SS0)x]\!]$. Applying the definition of semantics, $[\![\forall x.x = (SS0)x]\!] = 1$ if and only if for each assignment ξ of variables, identical to the one fixed in the model except for the value it assigns to x, it holds that $[\![x = (SS0)x]\!] = 1$. But, when $\xi(x) = 1$, $[\![x = (SS0)x]\!] = 0$ since the left side evaluates to 1 and the right side to 2.

Examples

Example 8.7

Fix the signature of arithmetic, and consider the standard model of natural numbers. Consider $[\![\forall x.\exists y.x = (SS0)y]\!]$. Applying the definition of semantics, the formula holds if, for each assignment ξ of variables, identical to the one fixed in the model except for the value of x, it holds that $[\![\exists y.x = (SS0)y]\!] = 1$. In turn, this happens when there is an assignment ξ', identical to ξ except for the value of y, such that $[\![x = (SS0)y]\!] = 1$.

For each ξ as above, fix $\xi'(y) = x/2$, the integer division of x by 2. Whenever x is even, it is immediate to check that $[\![x = (SS0)y]\!] = 1$ holds. On the contrary, when x is odd $[\![x = (SS0)y]\!] = 0$ as the left side differs from the right.

It is evident that there is no possibility to find an assignment ξ' as above for every possible choice of ξ, so the initial formula is false.

Soundness

Definition 8.8 (Validity)

A formula A is *valid* or *true* in a Σ-structure $\mathscr{M}$ together with an interpretation ν of variables, when $[\![A]\!] = 1$.
A set of formulae is *valid* or *true* when each formula in the set is valid.

Theorem 8.9 (Soundness)

In any Σ-structure $\mathscr{M}$, for any interpretation ν of variables, which makes true the theory T and the assumptions in the finite set Δ, if A is the conclusion of a proof π from Δ in T, then A is valid.

Soundness

Proof. (i)

First, we observe that, by Definition 8.3, the connectives act in the Boolean algebra on $\{0,1\}$ with $0<1$, so the $\wedge$, $\vee$, $\neg$ operations are defined as in the truth-table semantics.

The proof is by induction on the structure of the proof π: we prove that the interpretation of the conclusion A is 1 when the interpretation of each G in the finite set of assumption Γ is 1:

- if π is a proof by assumption, then $A \in \Gamma$ and, by hypothesis $[\![A]\!] = 1$.
- if π is a proof by axiom, then $A \in T$, and, by hypothesis, $[\![A]\!] = 1$.
- if π is an instance of the Law of Excluded Middle, then $A \equiv B \vee \neg B$, and $[\![A]\!] = [\![B \vee \neg B]\!] = [\![B]\!] \vee \neg [\![B]\!] = 1$ by definition of complement.
- if π is an instance of $\top$-introduction, then $A \equiv \top$, so $[\![A]\!] = 1$.
- if π is an instance of refl, then $A \equiv \forall x : s.\, x = x$, so $[\![A]\!] = 1$ when $[\![x = x]\!] = 1$ for each possible evaluation of the variable x in $[\![s]\!]$. So, if x gets mapped to $e \in [\![s]\!]$, $(e,e) \in \{(z,z) : z \in [\![s]\!]\}$, so $[\![x = x]\!] = 1$ for any e.

↪

Soundness

↪ Proof. (ii)

- if π is an instance of sym, then $A \equiv \forall x\colon s.\forall y\colon s.x = y \supset y = x$, so $[\![A]\!] = 1$ when $[\![x = y \supset y = x]\!] = 1$ for each possible evaluation of the variables x and y in $[\![s]\!]$. So, if x gets mapped to $e_x \in [\![s]\!]$, and y to $e_y \in [\![s]\!]$, if $(e_x, e_y) \in \{(z,z)\colon z \in [\![s]\!]\}$, then $e_x = e_y$, thus $(e_y, e_x) \in \{(z,z)\colon z \in [\![s]\!]\}$, that is, $[\![x = y \supset y = x]\!] = 1$.
- if π is an instance of trans, then $A \equiv \forall x\colon s.\forall y\colon s.\forall z\colon s.x = y \wedge y = z \supset x = z$, so $[\![A]\!] = 1$ when $[\![x = y \wedge y = z \supset x = z]\!] = 1$ for each possible evaluation of the variables x, y, and z in $[\![s]\!]$. So, if x gets mapped to $e_x \in [\![s]\!]$, y to $e_y \in [\![s]\!]$, and z in $e_z \in [\![s]\!]$, if $(e_x, e_y) \in \{(z,z)\colon z \in [\![s]\!]\}$ and $(e_y, e_z) \in \{(z,z)\colon z \in [\![s]\!]\}$, then $e_x = e_y = e_z$, and thus $(e_x, e_z) \in \{(z,z)\colon z \in [\![s]\!]\}$, that is, $[\![x = y \wedge y = z \supset x = z]\!] = 1$.

↪

Soundness

↪ Proof. (iii)

- if π is an instance of fun, then $A \equiv \forall x_1 : s_1 \ldots \forall x_n : s_n . \exists! z : s_0 . z = f(x_1, \ldots, x_n)$, so $[\![A]\!] = 1$ exactly when z can be uniquely mapped into a value e_z in $[\![s_0]\!]$ so that $(e_z, [\![f]\!](e_{x_1}, \ldots, e_{x_n})) \in \{(z, z) : z \in [\![s]\!]\}$, which is evidently true for $e_z = [\![f]\!](e_{x_1}, \ldots, e_{x_n})$.
- if π is an instance of subst, then, by induction hypothesis, $[\![A[t/x]]\!] = 1$ and $[\![t = r]\!] = 1$, that is $[\![t]\!] = [\![r]\!]$. The conclusion follows by an easy induction on the structure of the formula A.
- if π is an instance of $\bot$-elimination, then, by induction hypothesis, $0 = [\![\bot]\!] = 1$. Thus, $[\![A]\!] = 1$ since interpretation is a total function.

↪

Soundness

↪ Proof. (iv)

- if π is an instance of $\wedge$-introduction, then $A \equiv B \wedge C$, and by induction hypothesis twice, $[\![B]\!] = 1$ and $[\![C]\!] = 1$. Thus, $1 = [\![B]\!] \wedge [\![C]\!] = [\![A]\!]$.
- if π is an instance of $\wedge_1$-elimination, then, by induction hypothesis, for some formula B, $[\![A \wedge B]\!] = [\![A]\!] \wedge [\![B]\!] = 1$. Thus, by definition of $\wedge$, $[\![A]\!] = 1$.
- if π is an instance of $\wedge_2$-elimination, then, by induction hypothesis, for some formula B, $[\![B \wedge A]\!] = [\![B]\!] \wedge [\![A]\!] = 1$. Thus, by definition of $\wedge$, $[\![A]\!] = 1$.
- if π is an instance of $\vee_1$-introduction, then $A \equiv B \vee C$ and, by induction hypothesis, $[\![B]\!] = 1$. So, by definition of $\vee$, $1 = [\![B]\!] \vee [\![C]\!] = [\![A]\!]$.
- if π is an instance of $\vee_2$-introduction, then $A \equiv B \vee C$ and, by induction hypothesis, $[\![C]\!] = 1$. So, by definition of $\vee$, $1 = [\![B]\!] \vee [\![C]\!] = [\![A]\!]$.
- if π is an instance of $\vee$-elimination, then, by induction hypothesis, for some formulae B and C, $[\![B \vee C]\!] = [\![B]\!] \vee [\![C]\!] = 1$, if $[\![B]\!] = 1$ then $[\![A]\!] = 1$, and if $[\![C]\!] = 1$ then $[\![A]\!] = 1$. By definition of $\vee$, either $[\![B]\!] = 1$, thus $[\![A]\!] = 1$, or $[\![C]\!] = 1$, thus $[\![A]\!] = 1$.

↪

Soundness

↪ Proof. (v)

- if π is an instance of $\supset$-introduction, then $A \equiv B \supset C$ for some formulae B and C. By induction hypothesis, if $[\![B]\!] = 1$ then $[\![C]\!] = 1$. So, by definition of $\supset$, $[\![A]\!] = 1$.
- if π is an instance of $\supset$-elimination, then, for some formula B, by induction hypothesis twice, $[\![B \supset A]\!] = 1$ and $[\![B]\!] = 1$. By definition of $\supset$, $[\![A]\!] = 1$.
- if π is an instance of $\neg$-introduction, then $A \equiv \neg B$ for some formula B. So, by induction hypothesis, if $[\![B]\!] = 1$ then $0 = [\![\bot]\!] = 1$. Thus, $[\![\neg B]\!] = 1$ as, either $[\![B]\!] = 0$, or $0 = 1$.
- if π is an instance of $\neg$-elimination, then $A \equiv \bot$ and, by induction hypothesis twice, $[\![\neg B]\!] = 1$ and $[\![B]\!] = 1$. So, by definition of complement, $0 = 1$. Thus, $0 = [\![A]\!] = 1$.

↪

Soundness

↪ Proof. (vi)

- if π is an instance of $\forall$-introduction, then $A \equiv \forall x\colon s.\, B$, and, by induction hypothesis, $[\![B]\!] = 1$ for every evaluation of variables which makes the assumptions true. But, since $x\colon s$ does not appear free in any assumption, $[\![B]\!] = 1$ for any way we may evaluate x in $[\![s]\!]$, that is $[\![A]\!] = 1$.
- if π is an instance of $\forall$-elimination, then $A \equiv B[t/x]$, and, by induction hypothesis, $[\![\forall x\colon s.\, B]\!] = 1$. So, in particular, when x evaluates to $[\![t]\!]$, $[\![A]\!] = [\![B[t/x]]\!] = 1$.

↪

Soundness

↪ Proof. (vii)

- if π is an instance of $\exists$-introduction, then $A \equiv \exists x\colon s.B$, and, by induction hypothesis, $[\![B[t/x]]\!] = 1$. So, the evaluation of variable ξ_s which is the same as ν_s except for $\xi_s(x) = [\![t]\!]$ makes A valid.
- if π is an instance of $\exists$-elimination, then, by induction hypothesis, $[\![\exists x\colon s.B]\!] = 1$ and, if $[\![B]\!] = 1$, then A is valid. But, $[\![\exists x\colon s.B]\!] = 1$ means that there is way to evaluate x in $[\![s]\!]$ which makes B valid. Applying this evaluation of variables to the second induction hypothesis, we get that A is valid. □

References

The interpretation of formulae, as illustrated in this lesson, has been formalised first by Alfred Tarski. This is a classical definition, and it can be found in most textbooks.

The notion of model, that is, a Σ-structure which satisfies all the axioms in a theory, is analysed in depth in the branch of Logic called *model theory*. A standard reference is *C.C. Chang* and *H.J. Keisler*, Model Theory, Studies in Logic and the Foundations of Mathematics, 3rd edition, Elsevier, (1990), ISBN 008088007X. Nevertheless, this text is quite dated, and an introduction to the basics of contemporary model theory can be found in *W. Hodges*, A Shorter Model Theory, Cambridge University Press, (1997), ISBN 0-521-58713-1.

The soundness theorem is a classical result and its proof can be found in most textbooks. Our treatment follows the already cited *John Bell* and *Moshé Machover*, A Course in Mathematical Logic, North-Holland, (1977), ISBN 0-7204-28440. It is worth comparing the proof in this lesson with the propositional proof using the truth-tables semantics.

Mathematical Logic

Lecture 9

Dr Marco Benini

marco.benini@uninsubria.it

Dipartimento di Scienza e Alta Tecnologia
Università degli Studi dell'Insubria

a.a. 2016/17

Syllabus

First order logic:

- Completeness

Strategy

The completeness theorem is difficult, both technically and conceptually.

The strategy to prove it is indirect:

- Suppose A is true in any model satisfying Γ. Then $\Gamma \cup \{\neg A\}$ has no model.
- We will show that any set of formulae Δ which is consistent, i.e., non allowing to derive a contradiction, has a model. This is proved by constructing a sufficiently big set Θ containing Δ which has enough information to synthesise a model for itself.
- So, $\Gamma \cup \{\neg A\}$ must be non consistent. Which means that $\Gamma \vdash A$.

We need to prove each step. And we will start from the end.

Consistency

Definition 9.1 (Consistent set)

Fixed a first-order signature, a set of formulae Γ on it is *consistent* when it does not happen that $\Gamma \vdash A$ and $\Gamma \vdash \neg A$ for any formula A in the language.

Definition 9.2 (Maximal consistent set)

Fixed a first-order signature, a set of formulae Γ on it is *maximal consistent* when it is consistent and for any other set Δ on the same language such that $\Gamma \subset \Delta$, Δ is not consistent.

It should be stressed that being maximal consistent is a property which is **not** invariant with respect to the language.

Consistency

Proposition 9.3

For any set of formulae Γ and any formula A,

- $\Gamma \cup \{\neg A\}$ *is not consistent if and only if* $\Gamma \vdash A$*;*
- $\Gamma \cup \{A\}$ *is not consistent if and only if* $\Gamma \vdash \neg A$*.*

Proof.

If $\Gamma \cup \{\neg A\}$ is non consistent, then $\Gamma \cup \{\neg A\} \vdash B$ and $\Gamma \cup \{\neg A\} \vdash \neg B$ for some B. So, by implication introduction, $\Gamma \vdash \neg A \supset B$ and $\Gamma \vdash \neg A \supset \neg B$. Since $\vdash (\neg A \supset B) \wedge (\neg A \supset \neg B) \supset A$ can be easily proved using the double negation law, see Example 2.6, it follows that $\Gamma \vdash A$.
Conversely, $\Gamma \cup \{\neg A\} \vdash A$ by hypothesis, and $\Gamma \cup \{\neg A\} \vdash \neg A$ by the assumption rule, so $\Gamma \cup \{\neg A\}$ is not consistent.
By the double negation law, $\Gamma \cup \{A\}$ is non consistent if and only if $\Gamma \cup \{\neg\neg A\}$ is non consistent, thus the second part follows from the first one. □

Consistency

The completeness theorem says that: if a formula A is true in every model of the theory Γ, then there is a proof of A from Γ.

Now, by Proposition 9.3, it suffices to prove that: if a formula A is true in every model of the theory Γ, then $\Gamma \cup \{\neg A\}$ is not consistent.

We notice that any super set of a set of non consistent formulae is non consistent, too. The idea we want to pursue is to construct a sufficiently rich super set of any consistent set that allows to build a model.

Consistency

Proposition 9.4

A set Γ is maximal consistent if and only if it is consistent and, for every formula A, either $A \in \Gamma$ or $\neg A \in \Gamma$.

Proof.

Suppose Γ is maximal consistent. Then it is consistent by definition. Also, suppose there is A such that $A \notin \Gamma$ and $\neg A \notin \Gamma$, then $\Gamma \cup \{A\}$ and $\Gamma \cup \{\neg A\}$ must be both non consistent by definition. Thus, by Proposition 9.3, $\Gamma \vdash \neg A$ and $\Gamma \vdash A$, making Γ non consistent, which is a contradiction.
Conversely, suppose $\Gamma \subset \Delta$. Then, there is $A \in \Delta$ such that $A \notin \Gamma$. So, by hypothesis, $\neg A \in \Gamma \subset \Delta$. Thus, $\Delta \vdash A$ and $\Delta \vdash \neg A$ by assumption. □

Corollary 9.5

If Γ is maximal consistent and $\Gamma \vdash A$ then $A \in \Gamma$.

Proof.

Otherwise $\neg A \in \Gamma$, thus $\Gamma \vdash \neg A$, making Γ non consistent. □

Closure of maximal consistent sets

Proposition 9.6

Let Γ be a maximal consistent set. Then the following facts hold:

1. $\top \in \Gamma$; $\bot \notin \Gamma$;
2. *if* $A \equiv r(t_1, \ldots, t_n)$ *then either* $A \in \Gamma$ *or* $\neg A \in \Gamma$;
3. *if* $\neg\neg A \in \Gamma$ *then* $A \in \Gamma$;
4. *if* $A \wedge B \in \Gamma$ *then* $A \in \Gamma$ *and* $B \in \Gamma$; *if* $\neg(A \wedge B) \in \Gamma$ *then* $\neg A \in \Gamma$ *or* $\neg B \in \Gamma$;
5. *if* $A \vee B \in \Gamma$ *then* $A \in \Gamma$ *or* $B \in \Gamma$; *if* $\neg(A \vee B) \in \Gamma$ *then* $\neg A \in \Gamma$ *and* $\neg B \in \Gamma$;
6. *if* $A \supset B \in \Gamma$ *then* $\neg A \in \Gamma$ *or* $B \in \Gamma$; *if* $\neg(A \supset B) \in \Gamma$ *then* $A \in \Gamma$ *and* $\neg B \in \Gamma$;
7. *if* $\forall x : s.A \in \Gamma$ *then* $A[t/x] \in \Gamma$ *for each term* $t : s$;
8. *if* $\neg(\exists x : s.A) \in \Gamma$ *then* $\neg A[t/x] \in \Gamma$ *for each term* $t : s$.

Proof. (i)

Since $\Gamma \vdash \top$ by truth introduction, $\top \in \Gamma$. Hence, since $\neg\top$ is equivalent to $\bot$, $\bot \notin \Gamma$. The condition on atomic formulae follows from Proposition 9.4. $\hookrightarrow$

Closure of maximal consistent sets

↪ Proof. (ii)

If $A \wedge B \in \Gamma$ then, by conjunction elimination, $\Gamma \vdash A$ and $\Gamma \vdash B$. So, by Corollary 9.5, $A \in \Gamma$ and $B \in \Gamma$. Moreover, by the De Morgan's Laws, $\neg(A \vee B)$ is equivalent to $\neg A \wedge \neg B$, so the required result follows. Also, since $\neg(A \supset B)$ is equivalent to $A \wedge \neg B$, the required result follows.
If $A \vee B \in \Gamma$ and $A \notin \Gamma$, it must be $\neg A \in \Gamma$. So it is immediate to see that $\Gamma \vdash B$, i.e., $B \in \Gamma$. Moreover, by the De Morgan's Laws, $\neg(A \wedge B)$ is equivalent to $\neg A \vee \neg B$, so the required result follows.
If $A \supset B \in \Gamma$ and $\neg A \notin \Gamma$, it must be $A \in \Gamma$. So it is immediate to see that $\Gamma \vdash B$, i.e., $B \in \Gamma$. Also, by the double negation law, $\Gamma \vdash \neg\neg A \supset A$, so, if $\neg\neg A \in \Gamma$, $A \in \Gamma$, too.
If $\forall x : s.A \in \Gamma$, by the forall elimination rule, $\Gamma \vdash A[t/x]$ for any term $t : s$. Thus, $A[t/x] \in \Gamma$. Also, since $\neg\exists x : s.A$ is equivalent to $\forall x : s.\neg A$, the required result follows. □

Closure of maximal consistent sets

Proposition 9.7

Let Γ be a maximal consistent set in a language with equality. Then the following facts hold:

1. $t = t \in \Gamma$ *for all terms* t;
2. *if* $t = r \in \Gamma$, *then also* $r = t \in \Gamma$;
3. *if* $t = r \in \Gamma$ *and* $r = u \in \Gamma$, *then also* $t = u \in \Gamma$;
4. *if* $t_i = r_i \in \Gamma$ *for each* $1 \leq i \leq n$, *then* $f(t_1, \ldots, t_n) = f(r_1, \ldots, r_n) \in \Gamma$ *for every* $f : s_1 \times \cdots \times s_n \to s_0$ *in the language;*
5. *if* $t_i = r_i \in \Gamma$ *for each* $1 \leq i \leq n$, *then* $p(t_1, \ldots, t_n) \supset p(r_1, \ldots, r_n) \in \Gamma$ *for every* $p : s_1 \times \cdots \times s_n$ *in the language.*

Proof.

Since all these equalities can be deduced from Γ applying the inference rules in an elementary way, by Corollary 9.5 the results follow. □

Closure of maximal consistent sets

Two evident conditions are lacking from Proposition 9.6:

- if $\exists x\colon s.A \in \Gamma$ then $A[t/x] \in \Gamma$ for some term $t\colon s$;
- if $\neg(\forall x\colon s.A) \in \Gamma$ then $\neg A[t/x] \in \Gamma$ for some term $t\colon s$.

In fact, the second condition is equivalent to the first one, since $\neg(\forall x\colon s.A)$ is equivalent to $\exists x\colon s.\neg A$.

The first condition is lacking simply because it does not hold for any maximal consistent set. Take the language with just equality, and let $U = \{u, v\}$. Consider the variable evaluation σ which maps every variable x in u. Take Ψ as the collection of true formulae on the model U under the evaluation σ. Evidently, Ψ is consistent, since it has a model. Moreover, for any formula A, either it is true or false in that particular model, so either $A \in \Psi$ or $\neg A \in \Psi$.
But $\exists x.\neg x = y$, with x and y distinct variables, is true, while $(\neg x = y)[t/x]$ is false for any term t because the only terms are variables and all of them are interpreted into the same element u.

Henkin sets

Definition 9.8 (Henkin set)

A set of formulae Γ in a language is a *Henkin set* when Γ is maximal consistent in that language and

- if $\exists x : s.A \in \Gamma$ then $A[t/x] \in \Gamma$ for some term $t : s$;
- if $\neg(\forall x : s.A) \in \Gamma$ then $\neg A[t/x] \in \Gamma$ for some term $t : s$.

Thus, Henkin sets form a proper subclass of maximal consistent sets, and they are the *right* objects to look at, as they contain enough information to construct a model for themselves.

Canonical model

Lemma 9.9

If Γ is a Henkin set, then there is a Σ-structure $\mathscr{M}$ together with an evaluation of variables σ which makes Γ true.

Proof. (i)

Let T be the set of terms in the language. Define $t \sim r$ when $t : s, r : s \in T$ and $t = r \in \Gamma$. By the properties of a Henkin set, see Proposition 9.7, $\sim$ is an equivalence relation. So, it induces a partition on T. Thus, we define $U = \{\{[t]_\sim : t : s \in T\}\}_{s \in S}$, grouping partitions by sort.

For each function symbol $f : s_1 \times \cdots \times s_n \to s_0$ in Σ,

$$[\![f]\!]([t_1]_\sim, \ldots, [t_n]_\sim) = [f(t_1, \ldots, t_n)]_\sim \ .$$

Notice how this definition is legitimate, since the class $[f(t_1, \ldots, t_n)]_\sim$ does not depend on the choice of the representatives $[t_1]_\sim$, ..., $[t_n]_\sim$, by a direct application of Proposition 9.7. ↪

Canonical model

↪ Proof. (ii)

For each relation symbol $p\colon s_1 \times \cdots s_n$ in Σ,

$$[\![p]\!] = \{([t_1]_\sim, \ldots, [t_n]_\sim) : p(t_1, \ldots, t_n) \in \Gamma\} \quad .$$

Again, this definition is legitimate since it does not depend on the choice of the representatives $[t_1]_\sim$, ..., $[t_n]_\sim$ by Proposition 9.7.

So, let $\mathscr{M}$ be the Σ-structure having U as its universe, and interpreting function symbols and relation symbols as above.

Define σ, the evaluation of variables, as $\sigma(x\colon s) = [x]_\sim$.

By induction on the structure of terms, we show that $[\![t]\!] = [t]_\sim$:

- if $t \equiv x\colon s$ is a variable, $[\![t]\!] = \sigma(x\colon s) = [t]_\sim$;
- if $t \equiv f(t_1, \ldots, t_n)$, $[\![t]\!] = [\![f]\!]([\![t_1]\!], \ldots, [\![t_n]\!])$, and, by induction hypothesis, $[\![t]\!] = [\![f]\!]([t_1]_\sim, \ldots, [t_n]_\sim) = [f(t_1, \ldots, t_n)]_\sim = [t]_\sim$.

↪

Canonical model

↪ Proof. (iii)

By induction on the structure of formulae, we show that, when $A \in \Gamma$, $[\![A]\!] = 1$, and when $\neg A \in \Gamma$, $[\![A]\!] = 0$.

- if $A \equiv \top$, then $A \in \Gamma$ and, by definition, $[\![A]\!] = 1$.
- if $A \equiv \bot$, then $\neg A \in \Gamma$ and, by definition, $[\![A]\!] = 0$.
- if $A \equiv p(t_1, \ldots, t_n)$, $[\![A]\!] = 1$ if and only if $([\![t_1]\!], \ldots [\![t_n]\!]) \in [\![p]\!]$, that is, $([t_1]_\sim, \ldots, [t_n]_\sim) \in [\![p]\!]$, and, by definition of the model, this happens exactly when $p(t_1, \ldots, t_n) \in \Gamma$, i.e., when $A \in \Gamma$. When $\neg A \in \Gamma$, being Γ maximal consistent, $A \notin \Gamma$, so $[\![A]\!] = 0$.
- if $A \equiv t = r$, $[\![A]\!] = 1$ exactly when $[\![t]\!] = [\![r]\!]$, which is equivalent to $[t]_\sim = [r]_\sim$, and by definition of the model, $t = r \in \Gamma$. Again, if $\neg t = r \in \Gamma$, being Γ maximal consistent, $t = r \notin \Gamma$, and $[\![A]\!] = 0$.
- if $A \equiv \neg B$, $[\![A]\!] = 1$ exactly when $[\![B]\!] = 0$, and, by induction hypothesis, this happens exactly when $B \notin \Gamma$. Conversely, if $A \notin \Gamma$, then $B \in \Gamma$, being Γ maximal consistent, so, by induction hypothesis, $[\![B]\!] = 1$, i.e., $[\![A]\!] = 0$.

↪

Canonical model

↪ Proof. (iv)

- if $A \equiv B \wedge C$, $[\![A]\!] = 1$ if and only if $[\![B]\!] = 1$ and $[\![C]\!] = 1$, but, by induction hypothesis, this happens exactly when $B \in \Gamma$ and $C \in \Gamma$. So, when $A \in \Gamma$, by Proposition 9.6, $B \in \Gamma$ and $C \in \Gamma$, thus $[\![A]\!] = 1$. On the contrary, when $\neg A \in \Gamma$, by Proposition 9.6, $\neg B \in \Gamma$ or $\neg C \in \Gamma$, and, being Γ maximal consistent, either $B \notin \Gamma$ or $C \notin \Gamma$. In both cases, $[\![A]\!] \neq 1$, so $[\![A]\!] = 0$.
- if $A \equiv B \vee C$, $[\![A]\!] = 1$ if and only if $[\![B]\!] = 1$ or $[\![C]\!] = 1$, but, by induction hypothesis, this happens exactly when $B \in \Gamma$ or $C \in \Gamma$. So, when $A \in \Gamma$, by Proposition 9.6, $B \in \Gamma$ or $C \in \Gamma$, thus $[\![A]\!] = 1$. On the contrary, when $\neg A \in \Gamma$, by Proposition 9.6, $\neg B \in \Gamma$ and $\neg C \in \Gamma$, and, being Γ maximal consistent, $B \notin \Gamma$ and $C \notin \Gamma$. In both cases, $[\![A]\!] \neq 1$, so $[\![A]\!] = 0$.
- if $A \equiv B \supset C$, $[\![A]\!] = 1$ if and only if $[\![B]\!] = 0$ or $[\![C]\!] = 1$, but, by induction hypothesis, this happens exactly when $\neg B \in \Gamma$ or $C \in \Gamma$. So, when $A \in \Gamma$, by Proposition 9.6, $\neg B \in \Gamma$ or $C \in \Gamma$, thus $[\![A]\!] = 1$. On the contrary, when $\neg A \in \Gamma$, by Proposition 9.6, $B \in \Gamma$ and $\neg C \in \Gamma$, and, being Γ maximal consistent, $B \in \Gamma$ and $C \notin \Gamma$. In both cases, $[\![A]\!] \neq 1$, so $[\![A]\!] = 0$.

↪

Canonical model

↪ Proof. (v)

- if $A \equiv \forall x\colon s.B$, $[\![A]\!] = 1$ exactly when, in whatever way $x\colon s$ is interpreted in U, $[\![B]\!] = 1$. Since U is composed by equivalence classes of terms, $x\colon s$ is interpreted in $[t]_\sim$ for any term $t\colon s$. This means that $[\![B[t/x]]\!] = 1$ in the σ evaluation of variables. By Proposition 9.6, when $A \in \Gamma$, $B[t/x] \in \Gamma$ for every term $t\colon s$, so, by induction hypothesis, $[\![B[t/x]]\!] = 1$ for any term $t\colon s$, thus $[\![A]\!] = 1$. Furthermore, when $\neg A \in \Gamma$, being Γ a Henkin set, there is a term $t\colon s$ such that $\neg B[t/x] \in \Gamma$, so, by induction hypothesis, $[\![B[t/x]]\!] = 0$, thus $[\![A]\!] = 0$.
- if $A \equiv \exists x\colon s.B$, $[\![A]\!] = 1$ exactly when, there is a way to interpret $x\colon s$ in U such that $[\![B]\!] = 1$. By definition of U, $x\colon s$ is interpreted in $[t]_\sim$ for some term $t\colon s$. This means that $[\![B[t/x]]\!] = 1$ in the σ evaluation of variables. Being Γ a Henkin set, when $A \in \Gamma$, $B[t/x] \in \Gamma$ for some term $t\colon s$, so, by induction hypothesis, $[\![B[t/x]]\!] = 1$, thus $[\![A]\!] = 1$. Also, when $\neg A \in \Gamma$, by Proposition 9.6, there is a term $t\colon s$ such that $\neg B[t/x] \in \Gamma$, so, by induction hypothesis, $[\![B[t/x]]\!] = 0$, thus $[\![A]\!] = 0$.

↪

Canonical model

↪ Proof. (vi)

Summarising, we have constructed a Σ-structure $\mathscr{M}$ and an evaluation of variables σ such that each formula $A \in \Gamma$ is true in $\mathscr{M}$ under the σ evaluation. □

Corollary 9.10

The $\mathscr{M}$ model has a universe which does not exceed the size of the collection of all terms.

Existence of Henkin sets

Proposition 9.11

Let Γ be a consistent set of formulae on the signature Σ. Then, there is a set of formulae Δ on a signature Σ', extending Σ with constants, such that Δ is a Henkin set and $\Gamma \subseteq \Delta$.

Proof. (i)

Warning: we anticipate some set theory here!

Let λ be the cardinality of the collection of terms on Σ. Let

$$C = \bigcup_{s \in S} \{c_i^s : s \mid i < \lambda\}$$

be a collection of symbols for constants, such that no $c_i^s : s$ appears in Σ. Let Σ' be Σ extended with the set of constants in C.

The collection of all formulae over Σ' is a set with cardinality λ, as it is easy to verify by cardinal arithmetic. So, it can be well-ordered in the sequence $\mathbb{S} = \{S_i : i < \lambda\}$ by means of an equivalent of the Axiom of Choice. ↪

Existence of Henkin sets

↪ Proof. (ii)

By transfinite induction on λ, we define for every $i \leq \lambda$ a set Γ_i of formulae such that

1. $\Gamma_j \subseteq \Gamma_i$ for every $j < i$;
2. Γ_i is consistent;
3. no more that i constant in C occur in Γ_i.

We pose $\Gamma_0 = \Gamma$. Condition (1) holds vacuously; (2) holds by hypothesis; (3) holds since no constant in C appears in Γ by definition.

If $i \leq \lambda$ is a limit infinite ordinal, we put $\Gamma_i = \bigcup_{j<i} \Gamma_j$. By definition, condition (1) holds. If $\Gamma_i \vdash A$ and $\Gamma_i \vdash \neg A$, then each proof uses only a finite subset of assumptions, Γ_i^A and $\Gamma_i^{\neg A}$. But every finite subset of Γ_i is contained in some Γ_j, with $j < i$, so there is $m < i$ such that $\Gamma_i^A \subseteq \Gamma_m$ and $\Gamma_i^{\neg A} \subseteq \Gamma_m$, thus $\Gamma_m \vdash A$ and $\Gamma_m \vdash \neg A$, contradicting the inductive assumption that Γ_m is consistent. So Γ_i must be consistent, proving (2). Finally, since (3) holds for any $j < i$, because of (1), it must hold also for i, proving (3) ↪

Existence of Henkin sets

↪ Proof. (iii)

If $i < \lambda$ is a successor ordinal, say $i = k+1$, we distinguish three cases:

- If $\Gamma_k \cup \{S_k\}$ is non consistent, then $\Gamma_i = \Gamma_k$, and the three conditions clearly hold by inductive hypothesis.
- If $\Gamma_k \cup \{S_k\}$ is consistent and S_k is not of the form $\exists x\colon s.A$ or $\neg\forall x\colon s.A$, then $\Gamma_i = \Gamma_k \cup \{S_k\}$. Evidently, the three conditions hold by inductive hypothesis and by construction of Γ_i.
- If $\Gamma_k \cup \{S_k\}$ is consistent and S_k has the form $\exists x\colon s.A$ or $\neg\forall x\colon s.A$, then, by (3), there is $c\colon s$ in C not occurring in Γ_k and S_k.
  So, $\Gamma_i = \Gamma_k \cup \{S_k, B[c/x]\}$ with $B \equiv A$ when $S_k \equiv \exists x\colon s.A$, and $B \equiv \neg A$ when $S_k \equiv \neg\forall x\colon s.A$. Clearly, (1) and (3) hold for Γ_i.
  Suppose Γ_i to be non consistent. Then, $\Gamma_k \cup \{S_k\} \vdash \neg B[c/x]$. Since c is new, it could be regarded as a variable free in the assumptions, so $\Gamma_k \cup \{S_k\} \vdash \forall x\colon s.\neg B$. If $S_k \equiv \exists x\colon s.A$, $B \equiv A$, thus $\Gamma_k \cup \{S_k\} \vdash \bot$ by exists-elimination. If $S_k \equiv \neg\forall x\colon s.A$, $B \equiv \neg A$, thus $\Gamma_k \cup \{S_k\} \vdash \bot$ since $\neg B$ is equivalent to A. In both cases, $\Gamma_k \cup \{S_k\}$ is non consistent, contradicting the assumption. Thus, Γ_i must be consistent. ↪

Existence of Henkin sets

↪ Proof. (iv)

Let $\Delta = \Gamma_\lambda$. By (1), $\Gamma = \Gamma_0 \subset \Delta$, and, by (2), Δ is consistent.

Let A be a formula on Σ' such that $A \notin \Delta$. Since $A \equiv S_k$ for some $k < \lambda$, Γ_{k+1} must not contain A, which means, by construction of the sequence of Γ_i's, that $\Gamma_k \cup \{A\}$ is non consistent, thus also $\Delta \cup \{A\}$ is non consistent. Therefore, Δ is maximal consistent.

If $\exists x : s.\, A \in \Delta$ then $\exists x : s.\, A \equiv S_k$ for some $k < \lambda$, so Γ_{k+1} contains $A[c/x]$ for some new constant $c : s$. Similarly, if $\neg\forall x : s.\, A \in \Delta$ then $\neg\forall x : s.\, A \equiv S_k$ for some $k < \lambda$, so Γ_{k+1} contains $\neg A[c/x]$ for some new constant $c : s$. Thus, Δ is a Henkin set. □

Completeness

Theorem 9.12

If Γ is a consistent set of formulae on a signature Σ, then Γ is true on a model whose universe has a cardinality less or equal than the cardinality of the formulae in the language on Σ.

Proof.

By Proposition 9.11, Γ can be extended to a Henkin set Δ. By Lemma 9.9, Δ, and thus Γ, has a model satisfying the cardinality constraints. □

Theorem 9.13 (Completeness)

If every model of Γ makes A true, then $\Gamma \vdash A$.

Proof.

Clearly, if every model of Γ makes A true, then $\Gamma \cup \{\neg A\}$ has no model.
Thus, by Theorem 9.12, $\Gamma \cup \{\neg A\}$ is non consistent.
Then, by Proposition 9.3, $\Gamma \vdash A$. □

References

The first completeness proof for first-order logic has been given by Kurt Gödel. The proof presented in this lesson follows the techniques introduced by Leon Henkin.

Our treatment follows *John Bell* and *Moshé Machover*, A Course in Mathematical Logic, North-Holland, (1977), ISBN 0-7204-28440.

Gödel's proof was his doctoral dissertation, and it is based on a obscure formalism. Henkin's proof is a substantial reorganisation of Gödel's proof, emphasising that it involves the construction of a model.

Mathematical Logic

Lecture 10

Dr Marco Benini

marco.benini@uninsubria.it

Dipartimento di Scienza e Alta Tecnologia
Università degli Studi dell'Insubria

a.a. 2016/17

Syllabus

First-order logic

- Compactness
- Löwenheim-Skolem Theorems
- A discussion on completeness

Compactness

Theorem 10.1 (Compactness)

For any set of formulae Γ, if every finite subset of Γ has a model, then Γ has a model too.

Proof.

By hypothesis, applying the Soundness Theorem 8.9, every finite subset of Γ is consistent.

Suppose Γ to be non consistent: then $\Gamma \vdash A$ and $\Gamma \vdash \neg A$. Since a finite number of assumptions occur in each proof, there are two finite subsets such that $\Gamma_1 \vdash A$ and $\Gamma_2 \vdash \neg A$. Consider $\Gamma_\omega = \Gamma_1 \cup \Gamma_2$. It is evidently finite and non consistent, leading to a contradiction. Thus, Γ must be consistent.

So, by Theorem 9.12, Γ has a model. □

Compactness

Proposition 10.2

Fix a language with a single sort. If a set of sentences S has arbitrarily large finite models, then it has an infinite model.

Proof.

Define $\tau_n = \exists x_1, \ldots, x_n. \bigwedge_{1 \leq i < j \leq n} x_i \neq x_j$. Clearly, τ_n holds in any model whose universe has at least n distinct elements.

Consider any finite subset $F \subseteq S \cup \{\tau_n : n \in \mathbb{N}\}$. Let $K = F \cap \{\tau_n : n \in \mathbb{N}\}$. Since F is finite, K is finite too, so $m = \max\{n : \tau_n \in K\}$ is defined. Thus, since S has arbitrarily large finite models by hypothesis, F must have a finite model larger than m.

Thus, by Theorem 10.1, $S \cup \{\tau_n : n \in \mathbb{N}\}$ has a model $\mathscr{M}$. Since τ_n must hold for every $n \in \mathbb{N}$, $\mathscr{M}$ must have more than n distinct elements in its universe, for every $n \in \mathbb{N}$, thus it must be infinite. □

Löwenheim-Skolem

A classical result in Model Theory is

Theorem 10.3 (Downward Löwenheim-Skolem)

Let T be a theory on the signature Σ with just one sort. If T has an infinite model of cardinality $\alpha \geq |T|$, then T has a model of any cardinality β such that $\max(|\Sigma|, \aleph_0) \leq \beta \leq \alpha$.

We are not going to prove it, since we do not want to develop the concept of 'being elementary equivalent' for models.

Here $\aleph_0 = |\mathbb{N}|$, the cardinality of natural numbers.

Löwenheim-Skolem

Corollary 10.4

Any consistent theory T (on a single sort) such that $|T| \leq \aleph_0$, has a model whose universe has cardinality at most $\aleph_0$.

Proof.

Being consistent, T has a model. Either T has an infinite model or it does not. In the latter case, the result is obtained.

In the former case, by Theorem 10.3 taking $\beta = \aleph_0$, the result follows. □

Notice how the Completeness Theorem 9.12 allows to prove a similar, but weaker result, since the model has the cardinality of the formulae on the language.

Löwenheim-Skolem

Theorem 10.5 (Upward Löwenheim-Skolem)

Let T be a theory on the signature Σ with just one sort. If T has a model of cardinality $\alpha \geq \aleph_0$, then T has a model of any cardinality $\beta \geq \max(\alpha, |\Sigma|)$.

Proof.

Fix any $\beta \geq \max(\alpha, |T|)$, and extend the signature Σ by adding β new constants k_i, $i < \beta$. Let $T' = T \cup \{k_i \neq k_j : i < j < \beta\}$. Clearly, T' is a theory on the extended signature.

Let $F \subseteq T'$ be any finite subset of T'. Since it contains only a finite number of axioms of the form $k_i \neq k_j$, F has a model, because the model for T, being infinite, allows to interpret those axioms, and makes true the other axioms in F.

Thus, by compactness, T' has a model $\mathscr{M}$ and it must contain at least β distinct elements. But, by Theorem 10.3, there is model having exactly β cardinality, by using the cardinality of $\mathscr{M}$ as an upper bound. □

Löwenheim-Skolem

Theorem 10.6 (Löwenheim-Skolem)

Let T be a theory on the signature Σ with just one sort. If T has a model of cardinality $\alpha \geq \aleph_0$, then T has a model of each cardinality $\beta \geq \max(|\Sigma|, \aleph_0)$.

Proof.

Immediate, by combining the upward and downward Löwenheim-Skolem theorems. □

Corollary 10.7

If T is a consistent theory on the signature Σ with just one sort, then either T has a finite model, or it has a model for any cardinality greater than $\max(|\Sigma|, \aleph_0)$.

Discussion

The Compactness Theorem 10.1 is a consequence of the completeness result. One of its consequences is Proposition 10.2.

Thus, it is **impossible** to write a first-order theory which captures the notion of having finite models only. In fact, any theory T, either has finite models with a limit on their cardinality, or it has at least an infinite model.

Hence, the compactness result reveals a first, intrinsic limit to what can be expressed in the first-order language.

Discussion

The Löwenheim-Skolem Theorems provide other limitations to what can be expressed in a first-order theory.

For example, Corollary 10.4 says that every 'effective' and consistent theory has a model whose cardinality is either finite or $\aleph_0$. Here, by 'effective' we mean really writable, thus, at least, with a finite or denumerable number of symbols.

As a concrete instance, we get that the theory of real numbers, as developed in any textbook of mathematical analysis, which can be formally rendered as a first-order theory, has a countable model, which is much smaller than $\mathbb{R}$.

Saying the same thing in another, provocative way, Mathematical Analysis does not speak about real (or complex) numbers. It speaks about an infinite set which is much smaller than $\mathbb{R}$ or $\mathbb{C}$. So small that it disregards most of the reals (or complex numbers), which play no role in Analysis.
[Analysts are greatly disturbed by this sentence, but, nevertheless it is true, when we regard Mathematical Analysis as a formal theory!]

Discussion

Of course, we are investigating formal first-order theories. In this respect, the Löwenheim-Skolem Theorems say that not only every 'effective' theory has a finite or countable model, but if it has an infinite model, it has a model of any infinite cardinality.

This has a deep impact. Consider, for example, a formal and effective theory of arithmetic. Natural numbers form an obvious model, and the theory is intuitively consistent. So, by Corollary 10.7, it has models of any infinite cardinality.

In other words, **without even writing the formal theory**, as far as we require it to be effective, we know that it does not capture only the model of natural numbers. It must have models for each cardinal above $\aleph_0$.

Compactness, again

In topology, *compactness* is the property of a topological space $\mathscr{S}$ which says that every open covering A of $\mathscr{S}$ contains a finite sub-collection C that covers $\mathscr{S}$.

The Compactness Theorem can be stated in topological terms, which make evident why the term 'compactness' is appropriate, but obscuring the logical meaning, and the use we could make of that result to construct models.

Compactness, again

Fix a signature Σ. Let $\mathscr{S}$ be the class of all the Σ-structures, that is, of all the possible models.

For any formula ϕ and for any theory T define

$$\mathrm{Mod}(\phi) = \{\mathfrak{M} \in \mathscr{S} : \mathfrak{M} \text{ makes true } \phi\} \quad ,$$

$$\mathrm{Mod}(T) = \{\mathfrak{M} \in \mathscr{S} : \mathfrak{M} \text{ is a model for } T\} \quad .$$

Clearly, for any $\mathfrak{M} \in \mathscr{S}$, there is a formula ψ such that $\mathfrak{M}$ makes it true: any formula with no free variables is either true or false, so one of them satisfies the requirement. Thus, $\mathfrak{M} \in \mathrm{Mod}(\psi)$.
Suppose $\mathrm{Mod}(\phi) \cap \mathrm{Mod}(\psi)$ is not empty. Then, since every Σ-structure in the intersection necessarily validates $\phi \wedge \psi$, $\mathrm{Mod}(\phi \wedge \psi) \subseteq \mathrm{Mod}(\phi) \cap \mathrm{Mod}(\psi)$.

Compactness, again

In topological terms, the class $\mathscr{S}$ can be equipped with a topology $\mathscr{T}$, whose basis is the collection $\{\mathrm{Mod}(\phi)\colon \phi \text{ is a formula on } \Sigma\}$.

Being elements in the basis of $\mathscr{T}$, the $\mathrm{Mod}(\phi)$ are *open* subsets.
But the complement of $\mathrm{Mod}(\phi)$ is $\mathrm{Mod}(\neg\phi)$ since the logic is classical, and each formula is either true or false in a model. So, topologically, $\mathrm{Mod}(\phi)$ is also a *closed* subset.

Now, $\mathrm{Mod}(T) = \bigcap_{\phi\in T} \mathrm{Mod}(\phi)$, since each $\mathfrak{M} \in \mathrm{Mod}(T)$ has to validate every formula $\phi \in T$.

Hence, $\{\mathrm{Mod}(T)\colon T \text{ is a consistent theory on } \Sigma\}$ is the family of all closed subset of $\mathscr{S}$.

Compactness, again

In general, let $A = \{A_i\}_{i \in I}$ be a family of subsets of X, for some set X. Then A has the *finite intersection property*, if every finite sub-collection $J \subseteq I$ has non-empty intersection $\bigcap_{i \in J} A_i$.
A standard result of general topology says

Proposition 10.8

A space X is compact if and only if any collection of closed subsets of X with the finite intersection property has non-empty intersection.

The Compactness Theorem says that, if every finite subset of a theory T has a model, then T has a model, and vice versa.

In the topological language, it means that $\mathrm{Mod}(T) = \bigcap_{\phi \in T} \mathrm{Mod}(\phi)$ is non-empty exactly when, for each finite $T' \subset T$, $\mathrm{Mod}(T') = \bigcap_{\phi \in T'} \mathrm{Mod}(\phi)$ is non-empty. Equivalently, posing $A = \{\mathrm{Mod}(\phi)\}_{\phi \in T}$, if A has the finite intersection property, then A has non-empty intersection.

Hence, the Compactness Theorem says that the space $\mathrm{Mod}(T)$ equipped with the $\mathscr{T}$ topology is compact.

Comparing models

Let Σ be a signature with just one sort, and let T be a theory.
We have seen that T may have more than one model.

This means that we have a way to distinguish models. From the outside of a theory, this is obvious. But, from the inside?

If $\mathfrak{M}$ and $\mathfrak{N}$ are both models for T, and they are distinct, we would like to find a formula δ in the language on Σ which holds in $\mathfrak{M}$ but is false in $\mathfrak{N}$.

The question is: can we always find such a formula?

Comparing models

Completeness is a property of a formal system which says that whatever is true in any model, it can be derived.

But there is an alternative notion of completeness which says

Definition 10.9 (Completeness)

A theory T on the signature Σ is *complete* if, for every sentence ϕ on the same language, either ϕ is true in any model of T, or $\neg\phi$ is true in any model of T.

Here, by 'sentence' we mean a first-order formula with no free variables. Hence, it does not depend on the interpretation of variables, which simplifies the analysis.

Also, we will write $T \models \phi$ to say, that every model of T makes ϕ true.

So, we have another question: are the two notions of completeness equivalent?

Comparing models

Example 10.10

The simplest example of complete theory is $\mathrm{Th}(\mathfrak{M}) = \{\phi : \phi \text{ is true in } \mathfrak{M}\}$ with $\mathfrak{M}$ any model on the signature Σ.

The key in the example is that, since we are working in classical logic, every sentence is either true or false in a model. So, given two models $\mathfrak{M}$ and $\mathfrak{N}$, we can compare the models by comparing $\mathrm{Th}(\mathfrak{M})$ and $\mathrm{Th}(\mathfrak{N})$. When these theories are different, we know the models are different, too. And there is at least one sentence $\delta \notin \mathrm{Th}(\mathfrak{M}) \cap \mathrm{Th}(\mathfrak{N})$, which can be used to distinguish the models.

But, when they are equal?

Comparing models

Actually, the answer is simple: if a model $\mathfrak{M}$ is infinite, then the theory $\mathrm{Th}(\mathfrak{M})$ must have models of any infinite cardinality beyond the size of the language, by Theorem 10.5.

If we take one of those models, call it $\mathfrak{N}$, whose size is greater than the cardinality of $\mathfrak{M}$, we know that these models are distinct.

Consider $\mathrm{Th}(\mathfrak{N})$: since $\mathfrak{N}$ validates each formula in $\mathfrak{M}$, it makes true $\mathrm{Th}(\mathfrak{M})$, that is, for every $\phi \in \mathrm{Th}(\mathfrak{M})$, it holds that $\phi \in \mathrm{Th}(\mathfrak{N})$.

Since every sentence ϕ in the language on Σ is either in $\mathrm{Th}(\mathfrak{M})$, or $\neg\phi \in \mathrm{Th}(\mathfrak{M})$, then $\mathrm{Th}(\mathfrak{M}) = \mathrm{Th}(\mathfrak{N})$.

So, we may have **different** models which are indistinguishable by what we can express in the language.

Our counterexample shows that the models are distinguishable because they have different cardinality.

Comparing models

The way we obtained this negative result, shows that the two notions of completeness are not equivalent. We may write a theory T which is not complete, e.g., consider the theory of orders and the formula saying that the order has a global minimum.

Then, the collection of formulae which are true in any model of T is not a complete theory. In the counterexample of orders, the formula $\exists m. \forall x. m \leq x$ is true on the model of natural numbers, while it is false in the model of integer numbers.

As a remark, which concludes this part of the course, the complex construction behind the completeness theorem is unavoidable: the deep reason is that, whatever proving strategy we may want to pursue, we have to construct a complete model, in the sense introduced in this lesson, in which truth coincides with provability. And we are really forced to 'saturate' the theory to obtain such a model, otherwise the counterexamples we have just shown, could not be constructed.

References

The notion of compactness is fundamental in model theory, since it allows to construct models of an infinite theory by considering only finite subsets of formulae. This fact turns out to be critical in many situations. A good starting reference is *W. Hodges*, A Shorter Model Theory, Cambridge University Press, (1997), ISBN 0-521-58713-1.

The exposition of Löwenheim-Skolem theorems follows *John Bell* and *Moshé Machover*, A Course in Mathematical Logic, North-Holland, (1977), ISBN 0-7204-28440, omitting the parts on elementary equivalence of models. The same holds for the link between the logical Compactness Theorem and topology.

A comprehensive text on model theory which is approachable, but contains many examples of the application of logic to other fields, is *David Marker*, Model Theory: An Introduction, Graduate Texts in Mathematics 217, Springer (2002), ISBN 0-387-98760-6.

Mathematical Logic

Lecture 11

Dr Marco Benini

marco.benini@uninsubria.it

Dipartimento di Scienza e Alta Tecnologia
Università degli Studi dell'Insubria

a.a. 2016/17

Syllabus

Set theory:

- Language
- Classes and sets
- Paradoxes
- Comparing sets
- Axioms

Language

The language of the theory of sets is the usual first order language with equality plus one additional symbol: $\in$. The corresponding signature is

$$\langle \{S\}; \emptyset; \{=: S \times S, \in: S \times S\} \rangle$$

Since there is a unique sort, we omit sort specifications from the syntax.

The intended meaning is that S stands for the collection of all possible sets, while $\in$ denotes membership. Notice how there are no objects apart sets in the universe.

Language

It is important to distinguish between *formal* set theory, which is the first order theory we are going to introduce, and *informal* set theory which is used to describe the formal theory.

Although the former intends to model the latter, the latter is assumed in the definition of the former. With this distinction in mind, we cannot say that set theory is constructed out of itself.

As we have already seen, set theory admits a countable model, so the collection of all sets, seen 'from the outside' has the same cardinality as the natural numbers. But looking 'from the inside', the collection of all sets is much bigger.

This is just one of the various phenomena we should expect when dealing with the formal theory.

Language

The basic language of set theory is very poor, so it is enriched via a number of definitions, which are universally quantified:

- x not equal to y, $x \neq y$ abbreviates $\neg x = y$;
- x not in y, $x \notin y$ abbreviates $\neg x \in y$;
- x is a subset of y, $x \subseteq y$ abbreviates $\forall z. z \in x \supset z \in y$;
- there is x in y such that A, $\exists x \in y. A$ abbreviates $\exists x. x \in y \wedge A$;
- for all x in y, A, $\forall x \in y. A$ abbreviates $\forall x. x \in y \supset A$;
- for some subset x of y, A, $\exists x \subseteq y. A$ abbreviates $\exists x. x \subseteq y \wedge A$;
- for every subset x of y, A, $\forall x \subseteq y. A$ abbreviates $\forall x. x \subseteq y \supset A$;
- there is at most one x such that A, $\exists^* x. A$ abbreviates $\forall x. \forall y. A \wedge A[y/x] \supset x = y$ where $y \notin \mathrm{FV}(A)$.

Classes and sets

Informally, a set is a collection of elements. Although this is very intuitive and helpful, the structure of a set is much more subtle and delicate.

We stipulate that collections of elements are called *classes*. This is part of the intended meaning of set theory. Sets, in the intended meaning, are classes which behave in a *regular* way.

As we will see, there are classes which cannot be sets, while all sets are also classes, in the intended meaning. Each formal set has an *extension*, which is the class representing the collection of its element in the intended model of the theory. But, a set is **not** its extension, although we would like to say the converse, that is, to every extension corresponds a unique set.

As we will see, sets will have properties not shared by classes, e.g., sets have a *cardinality*, while proper classes have not. These properties are what identify the *structure* of sets, and they are what we are allowed to use when proving properties of sets, or when using sets in our proofs.

Paradoxes

A very simple theorem we will be able to derive in set theory will be: for any formula A such that $x \notin \mathrm{FV}(A)$,

$$(\exists x.\, \forall y.\, (y \in x) = A) \supset (\exists! x.\, \forall y.\, (y \in x) = A) \quad .$$

It means that, when there is a set x whose members are exactly those making the formula A true, then the set x is uniquely defined. In other words, the property A *defines* the set x.

It is tempting to carry on this result by thinking that any formula A defines a set. This amounts to assume

$$\exists x.\, \forall y.\, (y \in x) = A$$

as an axiom schema. This schema is usually called the unrestricted *Comprehension Axiom* and it has been used to define sets by Gottlob Frege.

Paradoxes

Unfortunately, the unrestricted Comprehension Axiom is untenable, as shown by **Russell's paradox**: take $A \equiv y \notin y$. Then, by the axiom, we have $\exists x. \forall y. y \in x = y \notin y$, and, specialising, we obtain $\exists x. x \in x = x \notin x$, allowing to derive $\bot$, i.e., showing that set theory is non consistent.

There are many variants of this paradox: we are presenting its formal version. It is important to understand the key point: the collection of sets making A true is a class. To be a set, it has to show a 'reasonable' behaviour. In logical terms, a minimal reasonable behaviour is not to allow to derive a contradiction.

Thus, what the Russell's paradox tells is

- there are classes which are not sets;
- every formula uniquely identifies a class: the elements which make it true. This class may be *proper*, that is, not a set.

Paradoxes

Sets are a delicate concept. When we fix a universe which is a set, and we do mathematics within that universe, we do not see the problems sets pose. But when we consider the totality of sets, things change.

Consider the following reasoning:

1. Let $X = \{x : x \in x \supset Y\}$
2. Assume $x = X$, then $(x \in x \supset Y) = (X \in X \supset Y)$
3. Thus, $X \in X$ is equivalent to $X \in X \supset Y$
4. So, an immediate deduction yields $X \in X \supset Y$ because this is equivalent to $X \in X \supset (X \in X \supset Y)$
5. the other way around, $(X \in X \supset Y) \supset X \in X$, so, by the previous step, we can deduce $X \in X$
6. Therefore, Y holds

Since Y can be any formula, fix $Y \equiv \bot$ and set theory becomes non-consistent. This is known as **Curry's paradox**, and step 3 is the wrong part, since it assumes X to be a set.

Paradoxes

Also, reasoning with arbitrary sets can be counter-intuitive. The **Sorites' paradox** can be adapted to provide an example:

1. Let S be a set
2. If $S = \emptyset$, stop
3. Otherwise, pick some element $x \in S$, and eliminate it,
4. obtaining a new S equal to $S \setminus \{x\}$
5. then iterate

The question is 'does the above procedure stop?'. The non-obvious answer is 'it depends'.

If we allow just a finite time, it does not. But if we allow a sufficiently large infinite time, which is linear and with an end-point, and we assume that picking an element can be done, then it stops.

But both assumptions are not immediate. In fact, we can drop them, and still we have a notion which behaves like a set. Or, conversely, we can assume them, and show that we must admit very bizarre entities as sets.

What is in the background here, is the **Axiom of Choice**.

Paradoxes

Sets and properties, as already seen, are linked, but different. Consider, for example, the **hyper-game paradox**. Let G be the collection of all games which can be played by two players by making successive alternate moves. A game in G is said to be *finite* if, in whatever way the players move, the game terminates after a finite number of steps. When a game is not finite, it is said to be *infinite*.

Take tic-tac-toe: it must end at most after 9 moves, so it is a finite game.

Define the *super-game* as the game in which the first player chooses a game $g \in G$, and then the second player starts playing g.

Is the super-game finite?

Paradoxes

Since the first player may choose an infinite game, the super-game is clearly infinite. So, define a variant: the *hyper-game* is played like the super-game, but the first player must choose a finite game.

Since the first player chooses a finite game g, then the hyper-game cannot take more than one move more than the moves to conclude g. But the moves to conclude g are always finite, so the hyper-game is finite.

Hence, the first player may choose the hyper-game as the game to play, and the second player may do the same. Forever. So the hyper-game is infinite.

Thus, the first player cannot choose the hyper-game, being infinite, and thus the hyper-game always terminate in a finite number of steps.

The problem here is that the collection of all finite games is a class, and we define the hyper-game as a particular element which depends on the whole class. This is something we want to do, but, as the paradox shows, it cannot be freely done with classes: a certain amount of 'regularity' in the class is needed to define an element which depends on it.

Comparing sets

Although many other paradoxes can be formed on sets, most of them require some knowledge that we have not yet explained.

A few facts, which seem to be paradoxical at the first sight, are of common use. And they are unavoidable.

Comparing two sets means to establish a correspondence between them. A function, mapping all the elements of one set in the element of another does not say much. But, when the function is bijective, we may think that the two sets are equal except for a renaming of the elements in their extensions. We write $A \cong B$ to indicate that there is bijective map between the sets A and B.

Intuitively, a set A is smaller than a set B when it can be embedded into B modulo a renaming: formally, this intuition is modelled by the existence of an injective function $A \to B$. Symmetrically, A is greater than B when there is a surjective function $A \to B$.

Comparing sets

This way of comparing sets is the standard, and it works as one expects when dealing with finite sets. But, on infinite sets, it reveals that sets are far more complex objects than we may imagine at a first sight.

Theorem 11.1 (Schröder-Bernstein)

If $f : A \to B$ is injective and $g : B \to A$ is injective then $A \cong B$.

Proof. (i)

Let $C_0 = A \setminus g(A)$ and, by induction, $C_{n+1} = \{g(x) : x \in D_n\}$ and $D_n = \{f(x) : x \in C_n\}$. Define

$$h(x) = \begin{cases} f(x) & \text{if } x \in C_n \text{ for some } n \\ g^{-1}(x) & \text{otherwise} \end{cases}$$

This definition makes sense, as $g^{-1}(x)$ is defined on $g(A)$. ↪

Comparing sets

↪ Proof. (ii)

Let $x, y \in A$. Suppose $h(x) = h(y)$: if $x \in C_m$ and $y \in C_k$ for some m and k, then $f(x) = f(y)$, so $x = y$ being f injective; if $x \notin C_n$ and $y \notin C_n$ for any n, then $g^{-1}(x) = g^{-1}(y)$, so $x = y$ being g injective; if $x \in C_m$ for some m and $y \notin C_n$ for any n, $f(x) = g^{-1}(y)$, so $(g \circ f)(x) = y$, that is, $y \in C_{m+1}$, which is impossible. Thus h is injective.

We must show that $h(A) = B$. Firstly, for any n and any $z \in D_n$, $z = f(x)$ for some $x \in C_n$, so, by definition, $z = h(x)$. Then, let $z \in B \setminus \bigcup_n D_n$. Evidently, by induction on n, $g(z) \notin C_n$ for any n, thus $h(g(z)) = g^{-1}(g(z)) = z$. So h is surjective. □

It is surprising how difficult is to prove this result, which is completely elementary in the finite case.

Comparing sets

Example 11.2

Let $P = \{2n : n \in \mathbb{N}\}$. Since $f : P \to \mathbb{N}$ such that $f(x) = x$ is injective, and $g : \mathbb{N} \to P$ such that $g(x) = 2x$ is injective, by Theorem 11.1 we conclude that $P \cong \mathbb{N}$.

In general, an infinite set A is such that it is possible to find a proper subset $B \subset A$ such that $A \cong B$. We can even use this property as a *definition* of being infinite.

Comparing sets

Example 11.3
$\mathbb{N} \times \mathbb{N} \cong \mathbb{N}$

Evidently, the function $f : \mathbb{N} \to \mathbb{N} \times \mathbb{N}$ mapping $x \mapsto (x, x)$ is injective.
Oppositely, the function $g : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ defined as
$g(x, y) = (x + y)(x + y + 1)/2 + y$ is injective, as it is easy to prove. Informally, it counts the pairs using diagonals which justifies the claim of being injective: the formal proof is just arithmetic.
Thus, by Theorem 11.1 the result follows.

This result can be generalised to arbitrary infinite sets, although the proof requires some technicalities.

A simpler result, which is immediately obtained by induction, is that $\mathbb{N}^k \cong \mathbb{N}$ for any $k > 0$.

Comparing sets

Example 11.4

The collection of finite sequences of naturals $\mathbb{N}^* \cong \mathbb{N}$

Obviously, the function $f : \mathbb{N} \to \mathbb{N}^*$ mapping $x \mapsto \{x\}$ is injective.
Oppositely, calling $g_n : \mathbb{N}^n \to \mathbb{N}$ the bijection from the Cartesian product of $n \geq 1$ copies of $\mathbb{N}$ to $\mathbb{N}$, we may define a function $h : \mathbb{N}^* \to \mathbb{N} \times \mathbb{N}$ by $h(\{x_i\}_{1 \leq i \leq n}) = (n, g_n(x_1, \ldots, x_n))$. For $n = 0$, let $h(\emptyset) = (0, 0)$.
Evidently, h is injective since g_n is, for each $n \geq 1$. So, the composition $g_2 \circ h$ is injective, and the result follows by Theorem 11.1.

Again, the result can be generalised to arbitrary infinite sets, essentially by the same proof.

Comparing sets

An application of what has been obtained till now to logic is immediate: let Σ be a signature with a finite number of symbols. Since the variables of sort s are in a bijective correspondence with $\mathbb{N}$, the collection of all variables is in bijection with $\mathbb{N}$.

Then, the sequences of symbols given by the function symbols, the parentheses, the commas, and the variables is in bijection with $\mathbb{N}$. So, the collection of all terms on Σ, being an infinite subset of that set, is in bijection with $\mathbb{N}$, too.

Analogously, the collection of all formulae on Σ, being an infinite subset of the collection of sequences of symbols of Σ plus a finite set of logical symbols, is in bijection with $\mathbb{N}$.

All these result can be easily extended to arbitrary signatures, using the generalised versions of the previous examples.

Comparing sets

Example 11.5

$\wp(\mathbb{N}) \ncong \mathbb{N}$.

This result, which specialises a famous Theorem by Cantor, says that the collection of subsets of $\mathbb{N}$ is **not** in bijection with $\mathbb{N}$. The proof is a classical masterpiece that introduces a technique called *diagonalisation*.

We can identify each subset $A \subseteq \mathbb{N}$ with its characteristic function $\chi_A : \mathbb{N} \to \{0,1\}$. Suppose that all these functions are in bijection with $\mathbb{N}$: then, there is a bijective function e which enumerates them. So, we have a sequence $\wp(\mathbb{N}) \cong \{\chi_{A_i}\}_{i \in \mathbb{N}}$ such that the i-th function is given by $e(i)$. Define a function $\Delta : \mathbb{N} \to \{0,1\}$ as $\Delta(x) = 1 - \chi_{A_x}(x)$. Thus Δ must appear somewhere in the sequence, i.e., $\Delta = \chi_{A_k}$ for some $k \in \mathbb{N}$. Which is impossible since $\chi_{A_k}(k) = \Delta(k) = 1 - \chi_{A_k}(k)$ and $\chi_{A_k} \in \{0,1\}$. Hence, the characteristic functions are not in bijection with $\mathbb{N}$, that is, $\wp(\mathbb{N}) \ncong \mathbb{N}$.

Again, this result can be generalised to any infinite set. As a side effect, since the functions $\mathbb{N} \to \{0,1\}$ are in evident bijection with the real interval $[0,1]$, we get that $\mathbb{R} > \mathbb{N}$ strictly. In other words, infinity is not unique!

Axioms: extensionality

Informally, a set is uniquely determined by its extension. This fact is captured by the following axiom:

Axiom (Extensionality)

$\forall x. \forall y. ((z \in x) = (z \in y)) \supset x = y.$

Proposition 11.6

If $x \notin \mathrm{FV}(A)$, *then* $\vdash (\exists x. \forall y. (y \in x) = A) \supset (\exists! x. \forall y. (y \in x) = A)$.

Proof.

The formal proof is easy, but long to write down. Essentially, if z is another set satisfying $\forall y. (y \in z) = A$, it must be that $x = z$ by extensionality. □

The content of the proposition is that, whenever the collection of the y's satisfying a formula corresponds to the extension of a set, it identifies a unique set.

Axioms: empty set

Axiom (Empty set)

$\exists x. \forall y. y \notin x.$

Since, by Proposition 11.6, the set x is unique, we will denote it by $\emptyset$, as usual. This axiom establishes that there is at least one set, the empty set.

Axioms: pairs

Axiom (Pair)

$\forall x. \forall y. \exists z. \forall u. (u \in z) = (u = x \vee u = y)$.

This axiom says that, given two elements x and y, we can form the set z whose extension contain exactly x and y. Again, we adopt the standard notation $\{x, y\}$, since, by extensionality, a pair set is uniquely identified.

Notice that, when $x = y$, we have *singletons*, $\{x\}$.

Axioms: union

Axiom (Union)

$\forall x.\exists y.\forall z.(z \in y) = (\exists u \in x.z \in u)$.

The axiom says that, given a set x, we can form another set y whose extension is the collection of elements in the members of x. Since, as usual, the set y is unique by extensionality, we adopt the standard notation $\bigcup x$ for it, or also, we write $\{z : \exists u \in x.z \in u\}$, or also $\bigcup_{u \in X} u$. When x is a pair $\{A, B\}$, we write $A \cup B$ for y.

Axioms: infinity

Axiom (Infinity)

$\exists x. \emptyset \in x \wedge \forall y. y \in x \supset y \cup \{y\} \in x.$

In general, we will write $\mathrm{Succ}(x)$ for $x \cup \{x\}$, and we will call it the *successor* of x. The axiom says that there is at least one set which is non empty, containing the empty set, and which is closed under the successor operation.

Not immediately, but it is possible to formally prove that there is a unique set that satisfies the axiom minimally, that is, its extension is minimal among all the collections containing the empty set and closed under the successor operation. This set is in biijection with the set of natural numbers. We will denote this minimal set as ω.

Axioms: power set

Axiom (Power set)

$\forall x.\exists y.\forall z.(z \in y) = (z \subseteq x)$.

The power set of x has as extension the collection of all the subsets of x. We will denote it as $\wp(x)$, or also $\{z : z \subseteq x\}$.

Working formally, by extensionality we get that, if $\wp(x) = x$, then $\forall y \in \wp(x).y \in x$, but $x \in \wp(x)$, so $x \in x$. Thus, as this behaviour is something we want to ban from our set theory, we want to introduce an axiom which prevents this phenomenon to happen. The consequence will be that $\wp(x) \neq x$ for every set x, thus proving the Cantor's Theorem.

Axioms: regularity

Axiom (Regularity)

$\forall x. x \neq \emptyset \supset \exists y \in x. \neg \exists z. z \in x \wedge z \in y.$

Similarly to extensionality, and differently from the preceding axioms, regularity states a property of all non empty sets, instead of providing a way to construct new sets. Precisely, it says that each non empty set x contains an element y which is disjoint from x.

It is a bit technical to show, and beyond the aims of this course, but the axioms prevents the construction of circular chains of membership, banning the existence of a set x satisfying $x \in x$, or $x \in y \in x$, ...

Thus, paradoxes like the hyper-game and Russell's cannot be constructed in the framework of formal set theory.

Axioms: separation

Axiom (Separation)

Let P be a formula such that $\mathrm{FV}(P) = \{u\}$, then
$\forall x. \exists y. \forall z. (z \in y) = (z \in x \wedge P[z/u])$.

Properly speaking, separation provides an *axiom schema*, i.e., a family of axioms, one for each possible instance of P.

It says that, given a set x, the collection of elements in x satisfying P is the extension of a set y.

An immediate application is the construction of intersection: $A \cap B$ is defined as the set formed by separation from A applying the property $P(u) = u \in B$.

Another immediate application is the construction of subsets: $\{x \in A \colon P\}$ is exactly the result of applying separation to A with the property P.

Axioms: replacement

Axiom (Replacement)

Let P be a formula such that $\mathrm{FV}(P) = \{x, y\}$*, then* $(\forall x. \exists! y. P) \supset \forall z. \exists u. \forall y. (y \in u) = (\exists x \in z. P)$.

It says that, whenever P behaves like a function mapping x to y, the image of any set x through P is a set.

Again, replacement is an axiom schema, whose instance are defined as soon as P is given.

Further definitions

With these fundamental definitions, together with their justifying axioms, we can easily define the usual operations on sets, like difference, Cartesian product, sequence, ...

The set theory developed so far is interesting by itself: it is called **ZF**, for Zermelo-Frænkel, its creators.

Although set theory is an important branch of mathematical logic, its development is far beyond the aim of this course, and involves some of the most stunning results of XX$^{\text{th}}$ century.

As a matter of fact, the collection of axioms we have shown is enough to develop most of elementary mathematics, although, in the following we will introduce another couple of axioms. In particular, the so-called *Axiom of Choice* has a special role, as it allows to prove some fundamental results in algebra, although it is also responsible for a few theorems which are really counter-intuitive, like the Tarski-Banach Theorem.

References

Probably, the best introductory text to set theory is *Paul Halmos*, Naive Set Theory. D. Van Nostrand Company, (1960). Reprinted by Springer-Verlag, (1974) ISBN 0-387-90092-6. Reprinted by Martino Fine Books (2011), ISBN 978-1-61427-131-4.

The axioms of set theory derive from the presentation in *Kenneth Kunen*, Set Theory: An Introduction to Independence Proofs, Studies in Logic and the Foundations of Mathematics 102, Elsevier, (1980), ISBN 0-444-86839-9. This book covers very advanced material, which lies far beyond the scope of the course. An alternative introduction can be found in *Jon Barwise*, Handbook of Mathematical Logic, Studies in Logic and the Foundations of Mathematics 90, North-Holland, (1977), ISBN 0-444-863888-5.

The theory **ZF** has been first proposed by Ernst Zermelo in 1908. Then, Abraham Fraenkel in 1921 pointed out that the original theory was not able to prove a number of natural properties of sets, so he and Thoralf Skolem in 1922 independently proposed an improved formulation, the one we introduced.

Mathematical Logic

Lecture 12

Dr Marco Benini

marco.benini@uninsubria.it

Dipartimento di Scienza e Alta Tecnologia
Università degli Studi dell'Insubria

a.a. 2016/17

Syllabus

Set theory:

- Ordinals
- Induction
- Arithmetic

Well orders

Definition 12.1

An order $\langle A; \leq \rangle$ is *total* when, for each pair $x, y \in A$, either $x \leq y$ or $y \leq x$.

Definition 12.2

An order $\langle A; \leq \rangle$ is a *well order* when every non empty subset $S \subseteq A$ has a minimum, i.e., there is $m \in S$ such that, for every $x \in S$, $m \leq x$.

Fixed a set A, it is always possible to add a relation to it so to make it an order, e.g., take $\leq$ to be equality. Also, it is immediate to define an order relation on A which makes it a total order, e.g., take $\leq$ to be the set $A \times A$. But it is not clear whether it is possible to define an order relation which makes it a well order.

However, a well order, as we will see soon, allows for an induction principle that is a very powerful instrument to reason about the set and its properties.

Ordinals

Definition 12.3

A set S is an *ordinal* if and only if $\langle S; \in \cup = \rangle$ is a total well order and, for each $x \in S$, $x \subseteq S$.

This definition is significant because it allows to prove

Proposition 12.4

Each ordinal S is totally well ordered by inclusion.

Proof.

Consider the structure $\langle S; \subseteq \rangle$. Clearly, $\subseteq$ forms an ordering relation. Also, being S an ordinal, for each $A, B \in S$, $A = B$, or $A \in B$, which implies, for all $x \in A$, $x \in B$ by transitivity, i.e., $A \subseteq B$, or $B \in A$, which implies, by the same argument, $B \subseteq A$. So, the structure is totally ordered.
Moreover, being S an ordinal, for each non empty $A \subseteq S$, there is $m \in A$ such that, for all $x \in A$, either $m = x$ or $m \in x$, that is, $m \subseteq x$. So, A is well ordered by inclusion, too. □

Ordinals

Proposition 12.5

If S is an ordinal and $x \in S$, then x is an ordinal.

Proof.

Immediate, since $x \in S$ implies $x \subseteq S$, being S an ordinal. □

Proposition 12.6

If U is a set of ordinals, then U is well ordered by inclusion.

Proof.

Consider the structure $\langle U; \subseteq \rangle$. It is evident that it forms an order. If $S \subseteq U$ is non empty, consider $\bigcap S = \{x : \forall y \in S. x \in y\}$. So $\bigcap S \subseteq y$, for all $y \in S$. Thus $\bigcap S$ is totally well ordered by $\in \cup =$ and, for each $x \in \bigcap S$, $x \subseteq y$, that is, for all $z \in x$, $z \in y$, so $\forall z \in x. z \in \bigcap S$, i.e., $x \subseteq \bigcap S$. Thus, $\bigcap S$ is an ordinal. Suppose $\bigcap S \notin S$. Then, $\bigcap S \in y$ for all $y \in S$ since $\bigcap S \subseteq y$ and both are ordinals. Thus $\bigcap S$ lies in the intersection of all y, in symbols, $\bigcap S \in \bigcap S$, contradicting the axiom of regularity. Thus $\bigcap S \in S$. □

Ordinals

Proposition 12.7

For each ordinal x, $x = \bigcup_{y \in x} y$.

Proof.

Immediate by Proposition 12.5. □

Proposition 12.8

The collection of all ordinals is not a set.

Proof.

Suppose $\mathrm{Ord} = \{x : x \text{ is an ordinal}\}$ is a set. Then it is immediate to check that Ord must be an ordinal. So $\mathrm{Ord} \in \mathrm{Ord}$, contradicting regularity. □

Admitting Ord to be a set generates a contradiction. This argument is called the *Burali-Forti* paradox.

Transfinite induction

Proposition 12.7 intuitively justifies

Principle 12.9 (Transfinite induction)

If P is a property, and, assuming that P holds for every ordinal less than α, we can prove that P holds for α, then P holds for any ordinal.

This principle can be relativised to all the ordinals less then some fixed ordinal β, leading to

Principle 12.10 (Transfinite induction)

If P is a property, and, assuming that P holds for every ordinal less than $\alpha < \beta$, we can prove that P holds for α, then P holds for any ordinal less than β.

Transfinite induction

We have to prove that transfinite induction is a sound principle, that is, it does not allow to derive false consequences from true statements.

Proposition 12.11

If P is a property, and, assuming that P holds for every ordinal less than α, we can prove that P holds for α, then P holds for any ordinal.

Proof.

Assume that, if $P(x)$ is true for every ordinal $x \in \alpha$, then $P(\alpha)$ holds. And, by contradiction, assume there is an ordinal β for which $P(\beta)$ is false.

Since β is an ordinal, it is well-ordered. Then, there exists the minimal ordinal $\gamma \leq \beta$ such that $P(\gamma)$ is false.

Being γ minimal, for every $x \in \gamma$, $P(x)$ is true. So, by hypothesis, $P(\gamma)$ holds, which contradicts the existence of γ, and thus, the existence of β. □

The relativised principle is an immediate corollary, by considering the property $\beta \leq x \vee P(x)$.

Transfinite induction

Since $\emptyset$ is an ordinal, and whenever x is an ordinal, its successor $x \cup \{x\}$ is an ordinal too, we can classify ordinals in three classes:

- the empty ordinal $\emptyset$;
- the *successor ordinals* x, such that there is an ordinal y for which $x = y \cup \{y\}$;
- the *limit ordinals* x, which are those ones not falling in the previous classes. These are characterised by $x = \bigcup_{y<x} y$.

It is worth remarking that the set of natural numbers is in bijection with ω, the ordinal containing $\emptyset$ and closed under the successor operation.

Transfinite induction

Principle 12.12 (Transfinite induction)

If P is a property and

- *if P holds for $\emptyset$;*
- *supposing P holds for an ordinal x, then P holds for the successor of x;*
- *supposing P holds for any ordinal $y < x$ with x a limit ordinal, then P holds also for x;*

we can conclude that P holds for any ordinal. Of course, as before, the principle can be relativised to the ordinals less than β.

Transfinite induction is a powerful instrument to reason about infinite sets: we already used it to prove the completeness theorem for first order logic.

Also, notice how the usual induction principle on natural numbers is equivalent to the transfinite induction principle relativised to ω.

Transfinite induction

Proposition 12.13

If α and β are ordinals, and $\alpha \cong \beta$, then $\alpha = \beta$.

Proof.

Let $f : \alpha \to \beta$ be a bijection between the ordinals.
Consider the property $P(x) \equiv \forall u \in \mathrm{Ord}. x \cong u \supset x = u$.

By transfinite induction on α, we have to show $P(\alpha)$ from the hypothesis that $P(x)$ holds for every $x \in \alpha$.
If $x \in \alpha$, then $f(x) \in \beta$, but also $x \subseteq \alpha$ since α is an ordinal. The restriction of the bijection f to x is a bijection, so $f(x) \cong x$, and by induction hypothesis, $f(x) = x$, thus $x \in \beta$. Hence, $\alpha \subseteq \beta$.

By transfinite induction on β, we have to show $P(\beta)$ from the hypothesis that $P(y)$ holds for every $y \in \beta$.
If $y \in \beta$, then $f^{-1}(y) \in \alpha$, but also $y \subseteq \beta$ since β is an ordinal. The restriction of the bijection f^{-1} to y is a bijection, so $f^{-1}(y) \cong y$, and by induction hypothesis, $f^{-1}(y) = y$, thus $y \in \alpha$. Hence, $\beta \subseteq \alpha$. □

Transfinite induction

Proposition 12.14

If $\langle A; \leq_A \rangle$ is a well-ordering, then there is a unique ordinal α such that $\langle A; \leq_A \rangle \cong \alpha$.

Proof.

Uniqueness follows by the previous proposition.

Let $\mathrm{pred}_A(a)$ be the initial segment of A up to a, that is, the well-order $\langle \{x \in A : x \leq_A a\}; \leq_A \rangle$ with the ordering restricted to the universe.
Define $B = \{a \in A : \exists x \in \mathrm{Ord}. \mathrm{pred}_A(a) \cong x\}$. Consider the formula $\phi(a,x) = (\mathrm{pred}_A(a) \cong x)$. Then, by the axiom of replacement and comprehension one forms $C = \{x : \exists a \in B. \phi(a,x)\}$. Thus, by comprehension we can form f, the function with domain B mapping a to x, the ordinal such that $\mathrm{pred}_A(a) \cong x$.
Now, C is an ordinal, since, for every $x \in C$, and for every $y \in x$, $y \in C$, that is, $C = \bigcup_{x \in C} x$. Also, f is a bijection, which preserves the order. So either $B = \mathrm{pred}_A(b)$ for some $b \in A$, thus $b \in B$ causing a contradiction, or $B = A$, in which case we have shown the result, with $\alpha = C$. □

Transfinite induction

What we have achieved is that any set which can be equipped with a well-order relation is isomorphic to an ordinal. Here, isomorphic has a double meaning:

- there is a bijection between the set and the ordinal;
- there is an invertible monotone function from the ordinal to the set.

Suppose that every set could be well-ordered. Then, the first sentence means that, up to renaming, all sets can be described as ordinals.

The second sentence says that transfinite induction can be applied to any set, modulo the monotone function, which is just a bijection, when we forget about the order.

Ordinal arithmetic

Definition 12.15 (Ordinal addition)

Let α and β be ordinals, then $\alpha+\beta$ is the unique ordinal such that there is $h\colon S\to\alpha+\beta$ biijective and monotone, i.e., such that $x\leq y$ in S implies $h(x)\leq h(y)$ in $\alpha+\beta$, where $S=\langle A\sqcup B;\leq\rangle$, the disjoint union of α and β, and $x\leq y$ if and only if x and y are both in α or in β, or $x\in\alpha$ and $y\in\beta$.

On finite ordinals, i.e., on natural numbers, it is just arithmetical addition. But, on infinite ordinals, it is not commutative. For example, $1+\omega=\omega$ but $\omega+1\neq\omega$ since, $\omega+1$ has a maximum, while ω has not.

The intuition one should keep in mind is that $\alpha+\beta$ is α followed by β.

Ordinal arithmetic

Proposition 12.16

Let α, β, and γ be ordinals. Then

1. $\alpha + (\beta + \gamma) = (\alpha + \beta) + \gamma$;
2. $\alpha + 0 = \alpha$;
3. $\alpha + 1 = Succ(\alpha)$;
4. $\alpha + Succ(\beta) = Succ(\alpha + \beta)$;
5. *if β is a limit ordinal, then* $\alpha + \beta = \bigcup_{\xi<\beta}(\alpha + \xi)$.

Proof. (i)

Reminding Proposition 12.13, it suffices to show that there is a bijective correspondence between the ordinals, to show their equality.

$\alpha + (\beta + \gamma) \cong \alpha \sqcup (\beta + \gamma) \cong \alpha \sqcup (\beta \sqcup \gamma) \cong (\alpha \sqcup \beta) \sqcup \gamma \cong (\alpha + \beta) \sqcup \gamma \cong (\alpha + \beta) + \gamma$, by applying the definitions of $+$ and $\sqcup$. Notice how all the bijections preserve the order.

$\alpha + 0 \cong \alpha \sqcup \emptyset = \alpha$. ↪

Ordinal arithmetic

↪ Proof. (ii)

$\alpha + 1 \cong \alpha \sqcup \{\alpha\} = \mathrm{Succ}(\alpha)$.

$\alpha + \mathrm{Succ}(\beta) \cong \alpha \sqcup \mathrm{Succ}(\beta) = \alpha \sqcup (\beta \sqcup \{\beta\}) = (\alpha \sqcup \beta) \sqcup \{\beta\} \cong (\alpha + \beta) \sqcup \{\beta\} \cong (\alpha + \beta) \sqcup \{\alpha + \beta\} \cong \mathrm{Succ}(\alpha + \beta)$.

$\alpha + \beta = \alpha + \bigcup_{\xi<\beta} \xi \cong \alpha \sqcup \bigcup_{\xi<\beta} \xi \cong \bigcup_{\xi<\beta} (\alpha \sqcup \xi)$. Notice how this bijection holds only when β is an infinite ordinal. Thus $\alpha + \beta \cong \bigcup_{\xi<\beta} (\alpha \sqcup \xi) \cong \bigcup_{\xi<\beta} (\alpha + \xi)$. Again, an essential part of the proof is that all the bijections preserve the order. □

Ordinal arithmetic

Definition 12.17 (Ordinal multiplication)

Let α and β be ordinals, then $\alpha\beta$ is the unique ordinal such that there is $h\colon S \to \alpha\beta$ biijective and monotone, where $S = \langle \bigsqcup_{i\in\beta} \alpha; \leq \rangle$ with $x \leq y$ in S when either $i < j$, $i,j \in \beta$ and $x \in \alpha_i$, $y \in \alpha_j$, or $x,y \in \alpha_i$ for some $i \in \beta$ and $x \leq y$ in α.

On finite ordinals, it is just arithmetical multiplication, but on infinite ordinals it is not commutative. For example, 2ω is the total order formed by ω copies of $0 < 1$. So, $2\omega = \omega$ by choosing $h(x) = 2i + x$ when $x \in 2_i$. On the contrary, $\omega 2 = \omega + \omega \neq \omega$ since there is a limit ordinal, ω, inside $\omega + \omega$, while there is none in ω.

The intuition behind ordinal multiplication is that $\alpha\beta$ is the ordinal consisting of the sequence composed by β copies of α.

Ordinal arithmetic

In general, it is immediate to show that $\alpha\beta \cong \langle \bigsqcup_{\xi\in\beta} \alpha; \leq_. \rangle$ with $(x,\xi_1) \leq_. (y,\xi_2)$ if and only if $\xi_1 < \xi_2 < \beta$ or $\xi_1 = \xi_2$ and $x \leq_\alpha y$.

Proposition 12.18

Let α, β, and γ be ordinals. Then

- $\alpha(\beta\gamma) = (\alpha\beta)\gamma$;
- $\alpha 0 = 0$;
- $\alpha 1 = \alpha$.

Proof.

Reminding Proposition 12.13, it suffices to show that there is a bijective correspondence between the ordinals, to show their equality.

$\alpha(\beta\gamma) \cong \langle \bigsqcup_{\xi<\beta\gamma} \alpha; \leq_. \rangle \cong \langle \bigsqcup_{\xi\in\bigsqcup_{\nu\in\gamma}\beta} \alpha; \leq_. \rangle \cong \langle \bigsqcup_{\xi\in\{(\beta,\nu):\, \nu\in\gamma\}} \alpha; \leq_. \rangle \cong$
$\langle \bigsqcup_{\xi\in\gamma} \bigsqcup_{\nu\in\beta} \alpha; \leq_. \rangle \cong \langle \bigsqcup_{\xi\in\gamma} (\alpha\beta); \leq_. \rangle \cong (\alpha\beta)\gamma.$

$\alpha 0 \cong \langle \bigsqcup_{\xi\in\alpha} \emptyset; \leq_. \rangle = 0.$

$\alpha 1 \cong \langle \bigsqcup_{\xi\in\alpha} \{\emptyset\}; \leq_. \rangle \cong \{(\emptyset,\xi) : \xi \in \alpha\} \cong \alpha.$ □

Ordinal arithmetic

We state, without proving

Proposition 12.19

Let α, β, and γ be ordinals. Then

- $\alpha Succ(\beta) = \alpha\beta + \alpha$;
- *If β is a limit ordinal,* $\alpha\beta = \bigcup_{\xi\in\beta}(\alpha\xi)$;
- $\alpha(\beta+\gamma) = \alpha\beta + \alpha\gamma$.

Notice how most of these properties do not commute when ordinals are infinite. For example, it is possible that $(\beta+\gamma)\alpha \neq \beta\alpha + \gamma\alpha$, in fact, $(1+1)\omega = 2\omega = \omega \neq 1\omega + 1\omega = \omega + \omega$.

References

The content of this lesson derives from the presentation in *Kenneth Kunen*, Set Theory: An Introduction to Independence Proofs, Studies in Logic and the Foundations of Mathematics 102, Elsevier, (1980), ISBN 0-444-86839-9.

Ordinals form, in a sense, the backbone of set theory, providing the main tool to prove properties of sets at large: transfinite induction.

Mathematical Logic

Lecture 13

Dr Marco Benini

marco.benini@uninsubria.it

Dipartimento di Scienza e Alta Tecnologia
Università degli Studi dell'Insubria

a.a. 2016/17

Syllabus

Set theory:

- Well orders
- Induction
- Cardinals
- Arithmetic
- Axiom of choice
- The continuum hypothesis
- What is a set?

Well orders, again

Definition 13.1 (Segment)

In an order $\langle E; \leq \rangle$, $S \subseteq E$ is a *segment* in E when, for every $x \in S$ and $y \in E$ such that $y \leq x$, $y \in S$.

Proposition 13.2

Let $\langle E; \leq \rangle$ be an order, and let A and B be two segments of E. Then, $A \cap B$, $A \cup B$ and every segment S of A or B are also segments of E. Finally, E itself and $\emptyset$ are segments of E.

Proof.

Immediate, by unfolding definitions. □

Well orders, again

Proposition 13.3

In a well ordering $\langle E; \leq \rangle$, any segment S of E such that $S \neq E$ has the form $S = S_e = \{x \in E : x < e\}$ for some $e \in E$.

Proof.

Consider a segment S of E with $S \neq E$. Then $E \setminus S$ is not empty, thus it has a least element e. So, for every $x \geq e$, $x \notin S$, otherwise it would be that $e \in S$, which is impossible. Hence, $E \setminus S = \{x \in E : x \geq e\}$, that is, $S = \{x \in E : x < e\}$. □

Notice how, if S is not empty, it has a least element m, so $S = \{x \in E : m \leq x < e\}$.

Well orders, again

Proposition 13.4

Let $\langle E; \leq \rangle$ be a well ordering. Then E^*, the set of segments of E, is well ordered by inclusion. In particular, the mapping $x \mapsto S_x$ from E to $E^* \setminus \{E\}$ is bijective, monotone and with a monotone inverse.

Proof.

Clearly, if $x, y \in E$ and $x \leq y$, then $S_x \subseteq S_y$, with $S_x = S_y$ if and only if $x = y$. Therefore, the mapping in the statement is an isomorphisms of orders, by Proposition 13.3. Thus, $\langle E^*; \subseteq \rangle$ is a well order, obtained by adding to $E^* \setminus \{E\}$ a maximal element. □

Transfinite induction, again

Proposition 13.5

Let $\langle E; \leq \rangle$ be a well ordering, and let $\mathscr{S}$ be a set of segments of E with the following properties:

1. *every union of segments belonging to $\mathscr{S}$ belongs to $\mathscr{S}$;*
2. *if $S_x \in \mathscr{S}$ then $S_x \cup \{x\} \in \mathscr{S}$.*

Then, every segment of E belongs to $\mathscr{S}$.

Proof. (i)

Suppose there are segments of E not belonging to $\mathscr{S}$. Then $E^* \setminus \mathscr{S}$ is not empty, and by Proposition 13.4, there is a minimal with respect to inclusion segment S of E such that $S \notin \mathscr{S}$.
If S does not have a greatest element then S is the union of the segments S' of S distinct from S. But these segments, by Proposition 13.2, are also segments of E, thus, by definition of S, they belong to $\mathscr{S}$. Thus, by the first property in the hypotheses of the theorem, $S \in \mathscr{S}$, a contradiction. ↪

Transfinite induction, again

↪ Proof. (ii)

If S has a maximal element m, then $S = S_m \cup \{m\}$. But S_m is a segment of S and $S_m \neq S$, so, by Proposition 13.2, S_m is a segment of E, thus, by definition of S, $S_m \in \mathscr{S}$. Thus, by the second property in the assumptions of the theorem, $S \in \mathscr{S}$, which is another contradiction.
So, $E^* \setminus \mathscr{S} = \emptyset$, that is, $S = E^*$. □

Theorem 13.6 (Transfinite induction)

Suppose $\langle E; \leq \rangle$ is a well ordering and let P be a property: if, for all $x, y \in E$ such that $y < x$, $P(y)$ holds, then $\forall x. P(x)$ holds.

Proof.

Let $\mathscr{S}$ be the set of segments S of E such that $\forall x \in S. P(x)$. Clearly, every union of segments belonging to $\mathscr{S}$ belongs to $\mathscr{S}$. If $S_x \in \mathscr{S}$, then $P(x)$ holds by hypothesis. Thus $S_x \cup \{x\} \in \mathscr{S}$. By Proposition 13.5, $\mathscr{S} = E^*$, so $E \in \mathscr{S}$, which proves the statement. □

Comparing sets, again

Definition 13.7

For any pair of sets A and B, $A \preceq B$ if and only if there is an injective function $A \to B$. Also, we write $A \approx B$ when there is a bijective function $A \to B$. Finally $A \prec B$ when $A \preceq B$ but $B \not\preceq A$.

Proposition 13.8

The relation $\preceq$ is reflexive and transitive, while $\approx$ is an equivalence relation.

Proof.

Since the identity function is bijective, $x \preceq x$ and $x \approx x$. Since the composition of injective (bijective) functions is injective (bijective), $\preceq$ ($\approx$) is transitive. Finally, since the inverse of a bijective function is bijective, $\approx$ is symmetric. □

Theorem 13.9 (Schröder-Bernstein)

If $A \preceq B$ and $B \preceq A$, then $A \approx B$.

Cardinals

Theorem 13.10

If $\langle E; \leq \rangle$ is a well ordering, then there is a unique ordinal α such that $\langle E; \leq \rangle \cong \alpha$. The notation $\cong$ means that there is a bijective and monotone function with a monotone inverse between the left and the right hands of the relation.

Proof. (i)

By Proposition 13.4, $\langle E; \leq \rangle \cong \langle E^* \setminus \{E\}; \subseteq \rangle$, and $\langle E^*; \subseteq \rangle$ is a well ordering. Take $S \in E^* \setminus \{E\}$, and proceed by transfinite induction:

- if $S = \emptyset$, then $\langle S; \subseteq \rangle$ is the ordinal 0;
- if S has a maximal element m, $S = S_m \cup \{m\}$, and, by induction hypothesis, $\langle S_m; \subseteq \rangle \cong \alpha_m$. Thus, $\langle S; \subseteq \rangle \cong \langle \mathrm{Succ}(\alpha_m); \subseteq \rangle$ in the obvious way. Uniqueness is evident. ↪

↪ Proof. (ii)

- if S has no maximum, $S = \bigcup_{S' \in E^*, S' \subset S} S'$ and, by inductive hypothesis, for each $S' \in E^*$ with $S' \subset S$, $\langle S'; \subseteq \rangle \cong \langle \alpha_{S'}; \subseteq \rangle$. Thus $\alpha_S = \bigcup_{S' \in E^*, S' \subset S} \alpha_{S'}$ is an ordinal by applying the definition. Furthermore, there is a function f from S to α_S that maps $x \in S$ to α_{S_x}. The function f is obviously injective, and it is surjective by Proposition 13.3. If $x \leq y$, $S_x \subseteq S_y$, and $\alpha_{S_x} \subseteq \alpha_{S_y}$, so f is monotone and its inverse is monotone, too. Thus, $\langle S; \subseteq \rangle \cong \langle \alpha_S; \subseteq \rangle$. Uniqueness is immediate, being f bijective.

Thus, by transfinite induction, $\langle E^* \setminus \{E\}; \subseteq \rangle \cong \alpha$ for some ordinal α which is unique. □

Cardinals

Definition 13.11 (Cardinality)

If the set A can be well ordered, $|A|$, the *cardinality* of A is the least ordinal α such that $A \approx \alpha$.

By Theorem 13.10, if A can be well ordered, it holds that $A \approx \alpha$ for some ordinal α which depends on the well ordering. Forming the set of ordinals $\{\alpha : A \approx \alpha\}$, it has a minimum, so the definition of cardinality is well-founded.

Definition 13.12 (Cardinal)

An ordinal α is a *cardinal* if and only if $a = |\alpha|$.

Equivalently, the ordinal α is a cardinal whenever, for all $\beta \in \alpha$, $\beta \not\approx \alpha$.

Proposition 13.13

Let α and β be ordinals. If $|\alpha| \leq \beta \leq \alpha$ then $|\alpha| = |\beta|$.

Proof.

Since $\beta \subseteq \alpha$, $\beta \preceq \alpha$ and $\alpha \approx |\alpha| \subseteq \beta$, so $\alpha \preceq \beta$. Then $\alpha \approx \beta$ by Theorem 13.9. Thus $|\alpha| \approx \alpha \approx \beta \approx |\beta|$. □

Proposition 13.14

If $n \in \omega$ then $n \not\approx n+1$ and, for every ordinal α, if $\alpha \approx n$, then $\alpha = n$.

Proof.

By induction on n it follows immediately that $n \not\approx n+1$. The second part is an instance of Proposition 13.13 by noticing that $|n| = n$. □

Cardinals

Corollary 13.15

Each $n \in \omega$ is a cardinal and ω is a cardinal.

Definition 13.16

A set A is *finite* if and only if $|A| < \omega$; A is *countable* if and only if $|A| \leq \omega$. *Infinite* means not finite, and *uncountable* means not countable.

Notice that, if $A \approx n \in \omega$, then A can be well ordered, so $|A|$ is defined. This is a general fact: when $A \approx \alpha$, with α an ordinal, then A can be well ordered by the relation which is the image of $\subseteq$ through the bijection $\alpha \to A$. Hence, $|A|$ is defined.

If A cannot be well ordered, which is possible in the framework described so far, A is both infinite and uncountable.

Cardinal arithmetic

Definition 13.17

Let α and β be cardinals. Then $\alpha \oplus \beta = |\alpha \sqcup \beta|$ and $\alpha \otimes \beta = |\alpha \times \beta|$.

Notice how cardinal addition and cardinal product are different from ordinal addition and product.

Proposition 13.18

Cardinal addition and product are associative and commutative operations with units.

Proof.

Since $\alpha \sqcup \beta \approx \beta \sqcup \alpha$ and $\alpha \times \beta \approx \beta \approx \alpha$, commutativity follows. Also, associativity derives from the corresponding property of $\sqcup$ and $\times$, up to $\approx$. It is immediate to check that 0 and 1 are the units of addition and multiplication, respectively. □

Cardinal arithmetic

Proposition 13.19

Let α and β be cardinals. Then

1. $|\alpha + \beta| = |\beta + \alpha| = \alpha \oplus \beta$;
2. $|\alpha\beta| = |\beta\alpha| = \alpha \otimes \beta$.

Proof.

Immediate, unfolding the definitions. □

Proposition 13.20

For $n, m \in \omega$, $n \oplus m = n + m$ and $n \otimes m = nm$.

Proof.

By induction on $m \in \omega$. □

Cardinal arithmetic

Proposition 13.21

Every infinite cardinal is a limit ordinal.

Proof.

If α is an infinite cardinal and $\alpha = \beta + 1$, since $1 + \beta = \beta$, $\alpha = |\alpha| = |\beta + 1| = |1 + \beta| = |\beta|$, a contradiction. □

Proposition 13.22

If α is an infinite cardinal, $\alpha \otimes \alpha = \alpha$.

Proof.

By transfinite induction on α. Assume the property holds for smaller cardinals. Then, for $\beta < \alpha$, $|\beta \times \beta| = |\beta| \otimes |\beta| < \alpha$, applying Proposition 13.20 when β is finite. Define a well ordering $\lhd$ on $\alpha \times \alpha$ by $(\beta, \gamma) \lhd (\delta, \varepsilon)$ if and only if $\max\{\beta, \gamma\} < \max\{\delta, \varepsilon\}$, or $\max\{\beta, \gamma\} = \max\{\delta, \varepsilon\}$ and $\beta < \delta$, or $\max\{\beta, \gamma\} = \max\{\delta, \varepsilon\}$ and $\beta = \delta$ and $\gamma < \varepsilon$. Each $(\beta, \gamma) \in \alpha \times \alpha$ has no more than $|(\max\{\beta, \gamma\} + 1) \times \times (\max\{\beta, \gamma\} + 1)| < \alpha$ predecessors in $\lhd$, so the ordinal ϕ such that $\phi \cong \langle \alpha \times \alpha; \lhd \rangle$ by Theorem 13.10 is such that $\phi \leq \alpha$, whence $|\alpha \times \alpha| \leq \alpha$. Clearly, $\alpha \preceq |\alpha \times \alpha|$, so $\alpha = |\alpha \times \alpha|$. □

Cardinal arithmetic

Corollary 13.23

Let α and β be infinite cardinals. Then $\alpha \oplus \beta = \alpha \otimes \beta = \max\{\alpha, \beta\}$.

Theorem 13.24 (Cantor)

For any set A, $A \prec \wp(A)$.

Proof.

Clearly, $A \preceq \wp(A)$ by the mapping $x \in A \mapsto \{x\}$. Suppose there is a bijective map $f : A \to \wp(A)$, and define $B = \{x \in A : x \notin f(x)\}$, which is a set by the Comprehension Axiom. Since $B \subseteq A$, $B \in \wp(A)$, thus there is a $y \in A$ such that $f(y) = B$, being f surjective. Now, if $y \in B$ then $y \in f(y) = B$, which is impossible. Conversely, if $y \notin B = f(y)$, then $y \in B$ by the definition of B, another contradiction. Thus, f cannot be surjective. □

Proposition 13.25

For every cardinal α, there is cardinal β such that $\alpha < \beta$ strictly.

Proof.

If α is finite, this is obvious: take $\beta = \alpha + 1$. If α is infinite, $\omega \leq \alpha$. Let

$$W = \{r \subseteq \alpha \times \alpha : r \text{ is a well order relation on } \alpha\} \ .$$

Since α is an ordinal $W \neq \emptyset$. Let

$$S = \{\phi \text{ ordinal} : \phi \cong \langle \alpha; r \rangle \text{ with } r \in W\} \ .$$

The set S exists by the Axiom of Replacement since ϕ is unique. Thus $\beta = \bigcup S$ is an ordinal which is strictly greater than any ordinal in it. In other words, β is a cardinal. Since $\alpha \in S$, $\alpha < \beta$. □

Hierarchy of cardinals

Definition 13.26

For any cardinal α, α^+ is the least cardinal strictly greater than α. We say that the cardinal β is a *successor* cardinal when $\beta = \alpha^+$ for some cardinal α. We say that the cardinal β is a *limit* cardinal when $\beta > \omega$ and β is not a successor cardinal.

Definition 13.27

By transfinite induction, we define the map $\aleph$ from ordinals to cardinals:

- $\aleph_0 = \omega$;
- $\aleph_{\alpha+1} = (\aleph_\alpha)^+$;
- for γ a limit ordinal, $\aleph_\gamma = \bigcup_{\alpha<\gamma} \aleph_\alpha$.

By transfinite induction on the ordinal α one shows

Proposition 13.28

Each $\aleph_\alpha$ is a cardinal, and every infinite cardinal equals $\aleph_\alpha$ for some α. Also, the map $\aleph$ is monotone, $\aleph_\alpha$ is a limit cardinal if and only if α is a limit ordinal, and $\aleph_\alpha$ is a successor cardinal exactly when α is a successor ordinal.

Hierarchy of cardinals

Proposition 13.29

$\aleph_1 \leq \wp(\aleph_0)$.

Proof.

By Proposition 13.28, $\aleph_0 < \aleph_1$. By Theorem 13.24, $\aleph_0 < \wp(\aleph_0)$. By definition, $\aleph_1$ is the least cardinal greater than $\aleph_0$, so $\aleph_1 \leq \wp(\aleph_0)$. □

This result can be easily extended to any ordinal α.

Since, although we are not going to prove this fact, the collection of functions from the cardinal α to 2, the finite cardinal composed by two distinct elements, has the same cardinality as $\wp(\alpha)$, the notation $2^\alpha = \wp(\alpha)$ is common.

Axiom of Choice

We have mentioned the Axiom of Choice many times. In most cases, we said that this principle allows to say that any set can be well ordered, or, equivalently, that any set is in bijection with a cardinal.

Axiom (Choice)

For any non empty family $\{X_i\}_{i\in I}$ of non empty sets such that $X_i \cap X_j = \emptyset$ for any $i,j \in I$, $i \neq j$, there exists a function $f : I \to \bigcup_{i\in I} X_i$ such that $f(i) \in X_i$ for every $i \in I$.

The meaning is that, whenever we are given such a family, we have the ability to make a choice that simultaneously pick an element from each set.

Although this principle seems very natural, it cannot be derived from the **ZF** set theory. So, when we adopt this axiom, we will speak of **ZFC**, the Zermelo-Frænkel set theory with the Axiom of Choice.

Axiom of Choice

As a matter of fact, when I, the index set of the family, is finite, the Axiom of Choice can be derived from **ZF**. But, when I is infinite, this is not possible.

Some important results in Mathematics require the Axiom of Choice to be proved: as a small collection of examples, take

- every non empty vector space has a base;
- every field has an algebraic closure, which is unique modulo isomorphisms;
- the notion of adjunction in category theory;
- the compactness theorem in first order logic.

Axiom of Choice

But, the Axiom of Choice allows to prove critical results, like the the Tarski-Banach theorem.

Its geometric form is: given a sphere S in the usual Euclidean space, it is possible to divide it into a finite set of pieces, so to obtain, using only rotations and translations, a reassembling of those pieces in two spheres both identical to S.

Of course, this seems to be impossible, since we consider pieces which are measurable, or, if you prefer, they possess a volume. On the other hand, if we take pieces, i.e., subspaces of the sphere for which the notion of volume is meaningless, the above composition becomes possible. In the proof, the pieces are constructed using the Axiom of Choice.

Axiom of Choice

There a number of equivalent formulation of the Axiom of Choice: the most common and useful ones are

- the Well Ordering Theorem
- the Zorn Lemma
- the Hartog's Theorem
- the Cartesian product of a family $\{X_i\}_{i \in I}$ of non empty sets, is non empty.

Well ordering theorem

Theorem 13.30 (Well ordering)

For any set X, $X \approx |X|$.

Proof.

By the Axiom of Choice, there is function $c\colon \wp(X) \setminus \{\emptyset\} \to \bigcup \wp(X) = X$, such that, for every non empty $S \subseteq X$, $c(S) \in S$.
By transfinite induction we define a bijection s between X and some ordinal α: assuming $s(\beta)$ has been defined for all $\beta \in \alpha$, if $X \setminus \{s(\beta)\colon \beta \in \alpha\} \neq \emptyset$, then $s(\alpha) = c(X \setminus \{s(\beta)\colon \beta \in \alpha\})$. We note that the construction must eventually stop, otherwise X would be in bijection with a proper class, the collection of all ordinals. And, moreover, s is a bijection, as it is immediate to see. By definition, $|X|$ is the least ordinal which is in bijection with X, and we know that there is one, α. □

Well ordering theorem

Assuming the Well Ordering Theorem as an axiom, we can prove the Axiom of Choice: let $\mathscr{F}$ be a non empty family of non empty, pairwise disjoint sets. Consider $\bigcup_{X \in \mathscr{F}} X$: by the Well Ordering Theorem, for each $X \in \mathscr{F}$, $X \approx I_X$ for some ordinal I_X, that is, there is $g_X : I_X \to X$ bijective.
Then, we can define a choice function $f : \mathscr{F} \to \bigcup_{C \in \mathscr{F}} X$ as $f(X) = g_X(\emptyset)$.

Zorn lemma

Theorem 13.31 (Zorn Lemma)

If $\langle X; \leq \rangle$ is a non empty order such that every proper ordered subset has an upper bound, then $\langle X; \leq \rangle$ contains a maximal element, i.e., an element which is not smaller than any other element in X.

Theorem 13.32 (Hartog)

If X and Y are two sets, it holds that either $|A| \leq |B|$ or $|B| \leq |A|$.

Although we are not going to prove these results, they shed some light to the meaning of the Axiom of Choice: in fact, they say that the notion of cardinality takes the usual, intuitive meaning, only when we assume that principle to hold.

For this reason, when no set theory is specified, usually **ZFC** is intended.

Continuum Hypothesis

Another axiom which is commonly considered in the theory of sets is the so-called *Continuum Hypothesis*:

Axiom (Continuum Hypothesis)

$\aleph_1 = 2^{\aleph_0}$.

It admits an obvious generalisation:

Axiom (Generalised Continuum Hypothesis)

$\aleph_{i+1} = 2^{\aleph_i}$ *for every ordinal* i.

Although the generalised Continuum Hypothesis implies the plain version, the converse does not hold. And, both the versions are independent from **ZFC**, that is, they cannot be proved from the axioms of **ZFC** nor it can be proved them to be false.

Continuum Hypothesis

While the Axiom of Choice justifies the intuitive notion of cardinality, the (generalised) Continuum Hypothesis is more technical and not easy to accept.

In fact, assuming the Continuum Hyothesis, the collection of all sets becomes a quite regular structure. On the contrary, assuming the Continuum Hypothesis to be false, the collection of all sets provides a very rich universe.

Intuition does not help: the effects of the Continuum Hypothesis are sensible for *large* sets, and the trade between regularity and wealth becomes difficult. In the common practice of higher set theory, which is far beyond the scope of this course, the Continuum Hypothesis is, generally, assumed not to hold, although some weaker regularity conditions may be considered.

What is a set?

As we said in the beginning, the notion of set is not simple.

The intuitive notion of a set as a collection of elements does not work, because of Russell's paradox. So, formal theories, like **ZFC**, have been introduced.

In those theories, a large number of principles, like the Axiom of Choice or the Continuum Hypothesis, are admissible but not provable: they are consistent with the theory, but also their negation is consistent with it.

So, *at least from the formal point of view*, we do not know exactly what is a set. We have a variety of structures (theories, if you prefer) that provide a reasonable notion of set. In some of these structures, we are able to prove results which are difficult to accept, like the Tarski-Banach Theorem. But, avoiding the principles underlying these structures, like the Axiom of Choice, we loose some basic, intuitive notion, like the cardinality of a set.

References

A nice reference to elementary set theory, which explains the nature of the Axiom of Choice with some detail is *P. Suppes*, Axiomatic Set Theory, Dover Publishing, (1972), ISBN 0-486-61630-4.

The classical text *Kenneth Kunen*, Set Theory: An Introduction to Independence Proofs, Studies in Logic and the Foundations of Mathematics 102, Elsevier, (1980) provides a more in-depth discussion, extending far beyond the limits of this course.

Another reference of interest is *N. Bourbaki*, Elements of Mathematics: Theory of Sets, Springer, (1968), ISBN 978-3-540-22525-6.

The continuum hypothesis is the main subject of the essays in *Paul J. Cohen*, Set Theory and the Continuum Hypothesis, Dover Publishing, (2008), ISBN 0-486-46921-2. This text contains the proof that the continuum hypothesis is independent from the other axioms of **ZFC**. Students should be warned that its content is advanced material.

Mathematical Logic

Lecture 14

Dr Marco Benini

marco.benini@uninsubria.it

Dipartimento di Scienza e Alta Tecnologia
Università degli Studi dell'Insubria

a.a. 2016/17

Syllabus

Computability:

- Motivation
- Recursive functions
- Main properties

Motivation

Computability theory is the branch of logic which studies the notion of 'computation'. Generally, it is considered in the borderline between mathematics and theoretical computer science, but, at least historically, it has been the part of logic from which computer science was born.

From a mathematical point of view, describing what can be really computed is an essential part of the XX^{th} century's mathematics. Consider the notion of algorithm and how fundamental it revealed in many fields.

For logicians, computability theory is an essential ingredient to understand the reasons behind constructive mathematics. But it is also the fundamental tool to prove the results about the limit of formal reasoning.

Computable functions

Computability theory aims at describing the functions $\mathbb{N} \to \mathbb{N}$ which can be effectively calculated.

We notice how the vast majority of functions from naturals to naturals cannot be calculated. In fact, if we think that calculation is a process which mechanically transforms the argument of a function in its result, we have to pose a few limits on this process:

- it must take a finite amount of time;
- it must operate on a finitely generated formal language;
- it must rely on a finite description of the process which precisely describes the steps to be performed.

At least, we have a language on a finite alphabet, which is used to describe the process. No matter how we interpret the language, we know that the set of all the possible procedures is contained in the collection of finite sequences of symbols in the alphabet. So, the cardinality of the language is at most $\aleph_0$, since the alphabet is finite. It is evident that it is at least $\aleph_0$ as we may write an infinite amount of procedures. But the cardinality of the set of functions from $\mathbb{N}$ to $\mathbb{N}$ is $2^{|\mathbb{N}|} = 2^{\aleph_0}$, which is strictly greater than $\aleph_0$. So, most functions are **not** computable.

Computable functions

There are many ways to describe computations. For our purposes, which are not aimed at studying computations, but rather using the computable functions to reason about what can be effectively proved inside a formal system, we will use *partial recursive functions*.

In fact, we admit a computation may not terminate, hence partial functions, in which non termination is modelled as the function being undefined for the non terminating input.

Instead of using some abstract machine which 'performs' the computation, we will directly define computable functions as the class of functions that can be written in a special form. Although it is not immediately clear that this class contains all the computable functions, it is best suited to application in logic.

Primitive recursive functions

Definition 14.1 (Primitive recursive functions)

A function $f : \mathbb{N}^k \to \mathbb{N}$ is *primitive recursive* when

1. f is the *zero* function $\underline{0}(n) = 0$ for all $n \in \mathbb{N}$;
2. f is the *successor* function $\underline{\mathrm{succ}}(n) = n + 1$ for all $n \in \mathbb{N}$;
3. f is a *projection* function $U_i^k(n_1, \ldots, n_k) = n_i$ with $k \geq 1$, $1 \leq i \leq k$;
4. f is obtained by *substitution*: if $g, h_0, \ldots, h_m$ are primitive recursive functions, $f(n_1, \ldots, n_k) = g(h_0(n_1, \ldots, n_k), \ldots, h_m(n_1, \ldots, n_k))$;
5. f is obtained by *primitive recursion*: if g and h are primitive recursive functions, $f(n_1, \ldots, n_k, 0) = g(n_1, \ldots, n_k)$ and $f(n_1, \ldots, n_k, m+1) = h(n_1, \ldots, n_k, m, f(n_1, \ldots, n_k, m))$.

It is clear that primitive recursive functions are computable. It is also evident that there are computable functions which are not primitive recursive: for example, the function undefined everywhere.

Primitive recursive functions

Example 14.2

The *identity* function $\mathrm{id}(x) = x$ is primitive recursive: $\mathrm{id} = U_1^1$.

Example 14.3

The constant function $\underline{k}(x) = k$ is primitive recursive. In fact, by induction on k, if $k = 0$, $\underline{0}$ is primitive recursive by definition; if $k = k' + 1$, $\underline{k} = \mathrm{succ} \circ \underline{k'}$ by substitution, and $\underline{k'}$ is primitive recursive by induction hypothesis.

Example 14.4

Addition, multiplication and exponentiation are primitive recursive.

$$n + 0 = n \qquad n \cdot 0 = 0$$

$$n + (m+1) = \mathrm{succ}\left(U_3^3(n, m, n+m)\right) \qquad n \cdot (m+1) = m + \underline{0}(n) + m \cdot n$$

$$n^0 = \underline{1}(n)$$

$$n^{m+1} = n \cdot \underline{1}(m) \cdot n^m$$

Notice how $0^0 = 1$, which sounds odd.

Primitive recursive functions

Example 14.5

The *predecessor* function, defined by

$$\mathrm{pred}(n) = \begin{cases} n-1 & \text{when } n > 0 \\ 0 & \text{otherwise} \end{cases}$$

is primitive recursive: $\mathrm{pred}(0) = \underline{0}(0)$, and $\mathrm{pred}(n+1) = U_1^2(n, \mathrm{pred}(n))$.

Example 14.6

The *recursive difference*, defined by

$$m \dot{-} n = \begin{cases} m-n & \text{if } m \geq n \\ 0 & \text{otherwise} \end{cases}$$

is primitive recursive: $m \dot{-} 0 = m$ and $m \dot{-} (n+1) = \mathrm{pred}(m \dot{-} n)$.

Primitive recursive functions

Example 14.7

The *absolute difference* $|m-n|$ is primitive recursive:

$$|m-n| = (m \dot{-} n) + (n \dot{-} m) \ .$$

Example 14.8

The *sign* function, defined by

$$\mathrm{sg}(n) = \begin{cases} 0 & \text{if } n = 0 \\ 1 & \text{otherwise} \end{cases}$$

is primitive recursive: $\mathrm{sg}(0) = \underline{0}(0)$, and $\mathrm{sg}(n+1) = U_1^2(n, \underline{1}(n))$.

Similarly, integer division, the remainder function, integer logarithm are primitive recursive.

Primitive recursive functions

There are functions which are computable but not primitive recursive.

Definition 14.9 (Ackermann)

The *Ackermann's function A* is defined as

$$\begin{aligned} A(m,0) &= m+1 \\ A(0,n+1) &= A(1,n) \\ A(m+1,n+1) &= A\big(A(m,n+1),n\big) \quad . \end{aligned}$$

To give an impression: $A(0,0)=1,\ A(1,1)=3,\ A(2,2)=7,\ A(3,3)=61,$ but

$$A(4,4) = 2^{2^{65536}} \quad .$$

The function $\mathbb{N} \to \mathbb{N}$ given by $n \mapsto A(n,n)$ can be shown to grow faster than any primitive recursive function, so it is **not** primitive recursive.

Partial recursive functions

Definition 14.10 (Partial recursive functions)

A partial function $f : \mathbb{N}^k \to \mathbb{N}$ is *recursive* when

1. f is the *zero* function $\underline{0}(n) = 0$ for all $n \in \mathbb{N}$;
2. f is the *successor* function $\underline{\mathrm{succ}}(n) = n + 1$ for all $n \in \mathbb{N}$;
3. f is a *projection* function $U_i^k(n_1, \ldots, n_k) = n_i$ with $k \geq 1$, $1 \leq i \leq k$;
4. f is obtained by *substitution*: if $g, h_0, \ldots, h_m$ are partial recursive functions, $f(n_1, \ldots, n_k) = g(h_0(n_1, \ldots, n_k), \ldots, h_m(n_1, \ldots, n_k))$;
5. f is obtained by *primitive recursion*: if g and h are partial recursive functions, $f(n_1, \ldots, n_k, 0) = g(n_1, \ldots, n_k)$ and $f(n_1, \ldots, n_k, m+1) = h(n_1, \ldots, n_k, m, f(n_1, \ldots, n_k, m))$;
6. f is obtained by *minimalisation*: if g is a partial recursive function, then $f(n_1, \ldots, n_k) = \mu m.\,(g(n_1, \ldots, n_k, m) = 0)$, with $\mu m.\,P(m) = m_0$ if and only if $P(m_0)$ holds, and, for all $m < m_0$, $P(m)$ does not.

We will speak of *recursive* functions when we will consider only computable total functions.

Partial recursive functions

Definition 14.11

Let S be a set and R a relation. The *characteristic functions* of S and R are given by

$$\chi_S(x) = \begin{cases} 1 & \text{if } x \in S \\ 0 & \text{if } x \notin S \end{cases}$$

$$\chi_R(x_1, \ldots, x_n) = \begin{cases} 1 & \text{if } (x_1, \ldots, x_n) \in R \\ 0 & \text{otherwise} \end{cases}$$

We say that S or R is *recursive* when χ_S or χ_R are total recursive functions. We say they are *primitive recursive* when the corresponding characteristic functions are.

Example 14.12

The relation $< \subseteq \mathbb{N} \times \mathbb{N}$ is primitive recursive: $\chi_<(n, m) = \mathrm{sg}(n \dot{-} m)$.

Partial recursive functions

Example 14.13

If P and Q are (primitive) recursive relations on $\mathbb{N}^k$, then so are $\neg P$, $P \wedge Q$, and $P \vee Q$.

$$\begin{aligned}
\chi_{\neg P}(x_1,\ldots,x_k) &= 1 \dot{-} \chi_P(x_1,\ldots,x_k) \\
\chi_{P\wedge Q}(x_1,\ldots,x_k) &= \chi_P(x_1,\ldots,x_k)\cdot\chi_Q(x_1,\ldots,x_k) \\
\chi_{P\vee Q}(x_1,\ldots,x_k) &= \mathrm{sg}\left(\chi_P(x_1,\ldots,x_k)+\chi_Q(x_1,\ldots,x_k)\right) \quad .
\end{aligned}$$

Example 14.14

Every finite set is primitive recursive.

Example 14.15

If R and S are primitive recursive subsets of $\mathbb{N}$, so are $\mathbb{N}\setminus R$, $R\cap S$, and $R\cup S$.

Partial recursive functions

Proposition 14.16

If $R(n_1, \ldots, n_k, m)$ is a recursive relation, then $f : \mathbb{N}^k \to \mathbb{N}$ defined by

$$f(n_1, \ldots, n_k) = \mu m.\, R(n_1, \ldots, n_k, m)$$

i.e., the least m such that $R(n_1, \ldots, n_k, m)$ holds, is partial recursive.

Proof.

Immediate by noticing that $f(n_1, \ldots, n_k) = \mu m.\, (\chi_{\neg R}(n_1, \ldots, n_k, m) = 0)$. □

Church-Turing Thesis

A function $f : \mathbb{N}^k \to \mathbb{N}$ is *computable* exactly when f is partial recursive.

Universal function

Theorem 14.17 (Enumeration)

There is a partial recursive function $e(x,y)$ that enumerates all the partial recursive functions, that is, defining $\phi_x(y) = e(x,y)$, $\{\phi_x\}_{x\in\mathbb{N}}$ is the collection of all the partial recursive functions.

Proof. (i)

In the first place, we notice that, since, for any $k \in \mathbb{N}$, $\mathbb{N}^k \cong \mathbb{N}$ and the bijection is computable, we may safely reduce to enumerate the computable functions $\mathbb{N} \to \mathbb{N}$.

Partial recursive functions can be coded as naturals:

- $[\underline{0}] = 2$;
- $[\underline{\text{succ}}] = 3$
- $[U_i^k] = 5 \cdot 17^k \cdot 19^i$;
- substitution:
  $[g(h_0(n_1,\ldots,n_k),\ldots,h_m(n_1,\ldots,n_k))] = 7 \cdot 17^{[g]} \cdot 19^{[h_0]} \cdot \ldots \cdot p_{7+m}^{[h_m]}$, with $\{p_i\}_{i\in\mathbb{N}}$ the sequence of prime numbers;

↪

Universal function

↪ Proof. (ii)

- primitive recursion: $[f] = 11 \cdot 17^{[g]} \cdot 19^{[h]}$;
- minimalisation: $[f] = 13 \cdot 17^{[g]}$.

The coding is injective, so invertible, thanks to the unique factorisation in primes of any natural number. Moreover, it is computable, and the inverse is computable, too. Precisely, the coding is primitive recursive, as it is immediate to check.

Defining $\underline{\perp}$ as the partial function which is everywhere undefined, we can invert the $[\_]$ coding:

$$\phi_n = \begin{cases} f & \text{if there is } f \text{ such that} [f] = n \\ \underline{\perp} & \text{otherwise} \end{cases}$$

Since $\underline{\perp}(x) = \mu m.(\underline{1}(x) = 0)$, the decoding is computable.

Then, $e(x, y) = \phi_x(y)$. It enjoys the enumeration property by construction. □

Universal function

Proposition 14.18

There is no $\{f_x\}_{n\in\mathbb{N}}$ of all total computable functions which admits an enumeration function $e(x,z) = f_x(z)$.

Proof.

Consider the function $h(x) = f_x(x) + 1$. It is total, since each f_x is.
Assume there is a recursive function e enumerating $\{f_x\}_{n\in\mathbb{N}}$. Then, $h(x) = e(x,x) + 1$, so h is recursive.
But h also occurs in $\{f_x\}_{n\in\mathbb{N}}$, so there is $k \in \mathbb{N}$ such that $f_k = h$.
Thus, $h(k) = e(k,k) + 1 = f_k(k) + 1 = h(k) + 1$, hence $0 = 1$, a contradiction. □

Universal function

Theorem 14.19

Let $m, n \geq 1$. Then, there is a computable function $S_n^m : \mathbb{N}^{m+1} \to \mathbb{N}$ such that

$$f_\alpha(x_1, \ldots, x_m, y_1, \ldots, y_n) = f_{S_n^m(\alpha, x_1, \ldots, x_m)}(y_1, \ldots, y_n) \ .$$

Although we will not prove the theorem, we want to remark its meaning: it shows that considering some arguments as parameters is an admissible operation in the computational world.

We can start the study of computable functions by considering an enumeration of them, which has a couple of properties: being computable, and satisfying the S_n^m theorem. Then

Theorem 14.20 (Turing, 1936)

There is a computable partial function $U : \mathbb{N}^2 \to \mathbb{N}$ such that $f_n(x) = U(n, x)$.

Such a function is called *universal*, and it is the first computer. But this is another story...

Fixed points

Theorem 14.21 (Kleene)

If f is a computable partial function, then exists $k \in \mathbb{N}$ for which $\phi_{f(k)} = \phi_k$ in any good enumeration of the partial recursive functions.

Proof.

Let $h(x) = \phi_x(x)$. This partial function is computable because it can be written as $h(x) = U(x,x)$. Then, $f \circ h$ is computable, too. So, $f \circ h = \phi_e$ for some $e \in \mathbb{N}$.

Therefore, $\phi_{f(h(e))} = \phi_{\phi_e(e)} = \phi_{h(e)}$. Thus $k = h(e)$ is the sought fixed point. □

References

Computability theory, also known as recursion theory is a major branch of mathematical logic. A very nice introductory text is *Barry Cooper*, Computability Theory, Chapman & Hall/CRC Mathematics, (2004), ISBN 1-58488-237-9.

This lecture is mainly based on that text.

Mathematical Logic

Lecture 15

Dr Marco Benini

marco.benini@uninsubria.it

Dipartimento di Scienza e Alta Tecnologia
Università degli Studi dell'Insubria

a.a. 2016/17

Syllabus

Constructive mathematics:

- Motivation
- Intuitionistic logic
- Syntax
- Expressive power

Motivation

Consider the following

Proposition 15.1

There are a and b irrational numbers such that a^b is rational.

Proof.

Let $a = b = \sqrt{2}$. Then $a^b = \sqrt{2}^{\sqrt{2}}$ is either rational or irrational. In the former case, the statement is proved, otherwise take $a = \sqrt{2}^{\sqrt{2}}$ and $b = \sqrt{2}$. Then $a^b = \left(\sqrt{2}^{\sqrt{2}}\right)^{\sqrt{2}} = \sqrt{2}^2 = 2$. □

This proof is correct, but still unsatisfactory: at the end, we don't know a pair of irrationals with the stated property. We have a choice between two candidate pairs but no way to decide which pair satisfies our requirement.

Motivation

On the contrary the following proof is different:

Proof.
Let $a = \sqrt{2}$ and $b = \log_2 9$. It is well known that a is irrational, but also b is. In fact, if $\log_2 9 = m/n$ for some $m, n \in \mathbb{N}$, then, by the properties of logarithm, $2^m = 9^n$, which is impossible, since the left-hand of the equality is even, while the right-hand is odd. But
$a^b = \sqrt{2}^{\log_2 9} = 2^{(\log_2 9)/2} = 2^{\log_2 3} = 3.$ □

Here, the statement says that there are two irrationals a and b such that a^b is rational, and the proof provides an evidence for this exhibiting such a pair.

Motivation

In general, we would like that any time we have to prove a statement of the form $A \vee B$ or $\exists x.P$, we are able to indicate which disjunct hold between A and B, or a value for x. And, we would like that these pieces of information lie in the proof.

More precisely, we would like to say that a proof for statements of this form would consist of an algorithm that indicate the true disjunct or *constructs* a value for x.

This attitude is perfectly reasonable, but comes with a price: we cannot use anymore axioms that directly violate the requirement. In particular, there is an axiom in the classical system that evidently violates the requirement.
In fact, the Law of Excluded Middle says that $A \vee \neg A$ for any formula A, but it provides no way to decide which of these mutually exclusive facts holds.
So, the Law of Excluded Middle must be rejected if we adopt a notion of *proof* as the one above.

Motivation

The Law of the Excluded Middle is essential in the first proof of Proposition 15.1: it avoids the need to decide whether $\sqrt{2}^{\sqrt{2}}$ is rational or not (in fact, it is not).

But rejecting the Law of Excluded Middle is not sufficient. There are a number of principles which pose problems.
For example, the Axiom of Choice. In one of its consequences, the already cited Tarski-Banach theorem, we can cut a sphere into a finite number of pieces so that we can reassembly two spheres identical to the original one. The proof 'constructs' the pieces using the Axiom of Choice. But any non-mathematician would call that result a miracle unless you show **how to** cut the original sphere and **how to** reassemble the pieces! And any mathematician would notice that the proof does not provide an **effective** way to calculate the shape of the pieces.

Motivation

In fact, what we would like to have is a logical system which allows to *calculate* the objects or the choices we have to make. In a sense, we are interested in systems where proofs are a sort of algorithm to construct the results implicit in their statements.

This attitude toward Mathematics is called *constructivism* and it produced a different kind of logical systems. In these systems, principles, like the Law of Excluded Middle, are rejected or accepted on the basis that they permit or deny the possibility to 'construct' the objects their statement imply to exists, or the possibility to make the choices required in the proofs.

There are many constructive systems, and many variations on the theme. Different philosophical foundations have been proposed to support the constructive approaches, and there are degrees of constructiveness in the logical system which claim themselves to adhere to these approaches.

An indisputable fact is that constructive mathematics had, have and, probably, will have a deep impact in the study of computability.

Intuitionistic logic

Among the many constructive system, *intuitionistic logic* has a special place. Historically, it has been the first formal attempt to capture in a formal system the original idea of a constructive approach to Mathematics. Practically, it is the simplest, most studied, and, in some sense, best understood system in this line of thought.

In the following we will introduce intuitionistic first-order logic, showing some of its main features. Differently from the study we pursued of classical systems, we will not prove every result and we will easily skip over some important parts: the field of constructive mathematics is wide, deep, and complex, and our objective is to show how and why a non-classical system could be of interest.

Syntax

Syntactically, intuitionistic logic is very similar to classical logic. In the propositional case, formulae are formed in exactly the same way. In the first-order case, terms and formulae are constructed identically.

The difference lie in the construction of proofs: the valid intuitionistic proofs are the classical proofs in natural deductions where the Law of Excluded Middle does not appear. In other words, the propositional calculus and the first-order calculus are identical to the corresponding classical calculi except that the Law of Excluded Middle is dropped.

Expressive power

Evidently, by definition, every proof $\pi : \Gamma \vdash_T A$ performed in the intuitionistic logic, i.e., without the Law of Excluded Middle, is also a valid classical proof.

So, we may think that intuitionistic logic is less expressive than classical logic: possibly, there are statement which are provable in the classical system, which cannot be proved in the intuitionistic system, because they use the Law of Excluded Middle in an essential way. On the contrary, every result which can be proved in an intuitionistic system is also valid in a classical system, because each intuitionistic proof is also a classical proof where there is no application of the Law of Excluded Middle.

In a sense, the above remark is correct. But, in another sense, it is not...

Expressive power

. . . since the ability to prove more, having an additional inference rule, may lead to prove more theories to be non consistent.

For example, Church Thesis in computability theory says that a function $\mathbb{N} \to \mathbb{N}$ is computable if and only if there is Turing machine computing it. If we say that every function we can write in arithmetic is computable, we get the so-called formal Church Thesis. It turns out that the formal theory of arithmetic plus formal Church thesis is a perfectly reasonable intuitionistic theory, which can be proved to be consistent with respect to (classical) arithmetic. On the contrary, the very same theory in classical logic turns out to be inconsistent.

The reason is simple: in classical logic it is possible to prove that a function exists which is not computable, by showing that it is impossible that it is computable. So, the formal Church thesis, which asserts that every function is computable, leads to a contradiction. In intuitionistic the proof of that function to be not computable cannot be carried on.

Expressive power

From another point of view, in a sense, every theorem in classical logic can be proved in intuitionistic logic, modulo a translation. The precise statement is as follows:

Definition 15.2

The *Gödel-Gentzen translation* is a map of formulae to formulae inductively defined as:

- $(\top)^N = \top$, $(\bot)^N = \bot$;
- for any A atomic, $(A)^N = \neg\neg A$;
- $(A \wedge B)^N = (A)^N \wedge (B)^N$;
- $(A \vee B)^N = \neg\left(\neg(A)^N \wedge \neg(B)^N\right)$;
- $(A \supset B)^N = (A)^N \supset (B)^N$;
- $(\forall x : s. A)^N = \forall x : s. (A)^N$;
- $(\exists x : s. A)^N = \neg\forall x : s. \neg(A)^N$.

Expressive power

Proposition 15.3

In classical logic, for any formula A, there is a proof $\pi : \vdash A = (A)^N$.

Proof. (i)

By induction on the formula A:

- $A \equiv \bot, \top$: $\vdash \bot \supset \bot$ and $\vdash \top \supset \top$ by implication introduction so $\vdash \bot = \bot$ and $\vdash \top = \top$.
- A is atomic:

$$\dfrac{\dfrac{\dfrac{[A]^1 \quad [\neg A]^2}{\bot}\,\neg\mathrm{E}}{\neg\neg A}\,\neg\mathrm{I}^2}{A \supset \neg\neg A}\,\supset\mathrm{I}^1 \qquad \dfrac{\dfrac{}{A \vee \neg A}\,\mathrm{lem} \quad \dfrac{\dfrac{[\neg\neg A]^2 \quad [\neg A]^1}{\bot}\,\neg\mathrm{E}}{A}\,\bot\mathrm{E} \quad [A]^1}{A}\,\vee\mathrm{E}^1}{\neg\neg A \supset A}\,\supset\mathrm{I}^2$$

↪

Expressive power

↪ Proof. (ii)

- $A \equiv B \wedge C$: by induction hypothesis there are $\vdash B = (B)^N$ and $\vdash C = (C)^N$, and $(A)^N = (B)^N \wedge (C)^N$, so

$$\dfrac{\dfrac{\dfrac{[B \wedge C]^1}{B} \wedge \mathrm{E}_1 \quad \dfrac{[B \wedge C]^1}{C} \wedge \mathrm{E}_2}{\dfrac{(B)^N \qquad (C)^N}{(B)^N \wedge (C)^N} \wedge \mathrm{I}}}{B \wedge C \supset (B)^N \wedge (C)^N} \supset \mathrm{I}^1$$

$$\dfrac{\dfrac{\dfrac{[(B)^N \wedge (C)^N]^1}{(B)^N} \wedge \mathrm{E}_1 \quad \dfrac{[(B)^N \wedge (C)^N]^1}{(C)^N} \wedge \mathrm{E}_2}{\dfrac{B \qquad C}{B \wedge C} \wedge \mathrm{I}}}{(B)^N \wedge (C)^N \supset B \wedge C} \supset \mathrm{I}^1$$

↪

Expressive power

↪ Proof. (iii)

- $A \equiv B \vee C$: by induction hypothesis there are $\vdash B = (B)^N$ and $\vdash C = (C)^N$, and $(A)^N = \neg\left(\neg(B)^N \wedge \neg(C)^N\right)$, so

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[(\neg(B)^N \wedge \neg(C)^N)]^3}{\neg(B)^N}\wedge E_1 \quad \dfrac{[(\neg(B)^N \wedge \neg(C)^N)]^3}{\neg(C)^N}\wedge E_2}{\vdots}}{\dfrac{[B]^2 \quad \neg B}{\bot}\neg E} \quad \dfrac{[C]^2 \quad \neg C}{\bot}\neg E}{[B \vee C]^1 \qquad \bot}\vee E^2}{\bot}\neg I^3}{\neg\left(\neg(B)^N \wedge \neg(C)^N\right)}\supset I^1}{B \vee C \supset \neg\left(\neg(B)^N \wedge \neg(C)^N\right)}$$

for the implicit proofs, see the $\neg$ case.

↪

Expressive power

↪ Proof. (iv)

$$\dfrac{\dfrac{\dfrac{}{B \vee \neg B}\,\text{lem} \quad \dfrac{[B]^1}{B \vee C}\,\vee\text{I}_1 \quad \dfrac{\dfrac{}{C \vee \neg C}\,\text{lem} \quad \dfrac{[C]^2}{B \vee C}\,\vee\text{I}_2 \quad \dfrac{\bot}{B \vee C}\,\bot\text{E}}{B \vee C}\,\vee\text{E}^2}{B \vee C}\,\vee\text{E}^1}{\neg\left(\neg(B)^N \wedge \neg(C)^N\right) \supset B \vee C}\,\supset\text{I}^3$$

$$\dfrac{\dfrac{\dfrac{[\neg B]^1 \quad [\neg C]^2}{\vdots \quad \vdots}}{\dfrac{\neg(B)^N \quad \neg(C)^N}{\neg(B)^N \wedge \neg(C)^N}\,\wedge\text{I}} \quad [\neg\left(\neg(B)^N \wedge \neg(C)^N\right)]^3}{}\,\neg\text{E}$$

for the implicit proofs, see the $\neg$ case. ↪

Expressive power

↪ Proof. (v)

- $A \equiv B \supset C$: by induction hypothesis there are $\vdash B = (B)^N$ and $\vdash C = (C)^N$, and $(A)^N = (B)^N \supset (C)^N$, so

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[(B)^N]^2}{\vdots}}{\dfrac{[B \supset C]^1 \quad B}{C}\supset E}}{\vdots}}{\dfrac{(C)^N}{\left((B)^N \supset (C)^N\right)}\supset I^2}}{(B \supset C) \supset \left((B)^N \supset (C)^N\right)}\supset I^1 \qquad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[B]^2}{\vdots}}{\dfrac{[((B)^N \supset (C)^N)]^1 \quad (B)^N}{(C)^N}\supset E}}{\vdots}}{\dfrac{C}{B \supset C}\supset I^2}}{\left((B)^N \supset (C)^N\right) \supset (B \supset C)}\supset I^1$$

↪

Expressive power

↪ Proof. (vi)

- $A \equiv \neg B$: by induction hypothesis there is $\vdash B = (B)^N$, so

$$\dfrac{\dfrac{\dfrac{\dfrac{[(B)^N]^2}{\vdots}}{\dfrac{[\neg B]^1 \quad B}{\bot}\,\neg\mathsf{E}}\,\neg\mathsf{I}^2}{\neg(B)^N}\,\supset\mathsf{I}^1}{\neg B \supset \neg(B)^N} \qquad \dfrac{\dfrac{\dfrac{\dfrac{[B]^2}{\vdots}}{\dfrac{[\neg(B)^N]^1 \quad (B)^N}{\bot}\,\neg\mathsf{E}}\,\neg\mathsf{I}^2}{\neg B}\,\supset\mathsf{I}^1}{\neg(B)^N \supset \neg B}$$

↪

Expressive power

↪ Proof. (vii)

- $A \equiv \forall x.B$: by induction hypothesis there is $\vdash B = (B)^N$, and $(A)^N \equiv \forall x.(B)^N$ so

$$\dfrac{\dfrac{\dfrac{[\forall x.B]^1}{B}\,\forall \mathsf{E}}{\vdots}}{\dfrac{(B)^N}{\forall x.(B)^N}\,\forall \mathsf{I}}\,\supset \mathsf{I}^1 \qquad (\forall x.B) \supset \left(\forall x.(B)^N\right)$$

$$\dfrac{\dfrac{\dfrac{[\forall x.(B)^N]^1}{(B)^N}\,\forall \mathsf{E}}{\vdots}}{\dfrac{B}{\forall x.B}\,\forall \mathsf{I}}\,\supset \mathsf{I}^1 \qquad \left(\forall x.(B)^N\right) \supset (\forall x.B)$$

↪

Expressive power

↪ Proof. (viii)

- $A \equiv \exists x.\, B$: by induction hypothesis there is $\vdash B = (B)^N$, and $(A)^N \equiv \neg\forall x.\, \neg(B)^N$ so

$$\dfrac{\dfrac{\dfrac{[\exists x.\, B]^1 \qquad \dfrac{\dfrac{[B]^2}{\vdots}}{(B)^N} \qquad \dfrac{[\forall x.\, \neg(B)^N]^3}{\neg(B)^N}\,\forall\mathrm{E}}{\bot}\,\neg\mathrm{E}}{\bot}\,\exists\mathrm{E}^2}{\neg\forall x.\, \neg(B)^N}\,\neg\mathrm{I}^3}{(\exists x.\, B) \supset \neg\forall x.\, \neg(B)^N}\,\supset\mathrm{I}^1$$

↪

Expressive power

↪ Proof. (ix)

$$\dfrac{\dfrac{}{(\exists x.B) \vee \neg(\exists x.B)}\,\text{lem} \quad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[(B)^N]^2}{\vdots}}{B}}{\exists x.B}\,\exists\text{I} \quad [\neg(\exists x.B)]^1}{\bot}\,\neg\text{E}}{\neg(B)^N}\,\neg\text{I}^2}{\forall x.\neg(B)^N}\,\forall\text{I} \quad [\neg\left(\forall x.\neg(B)^N\right)]^3}{\bot}\,\neg\text{E}}{\exists x.B}\,\bot\text{E} \quad [\exists x.B]^1}{\exists x.B}\,\vee\text{E}^1}{\neg\left(\forall x.\neg(B)^N\right) \supset \exists x.B}\,\supset\text{I}^3$$

Expressive power

Proposition 15.4

If $\pi : \Gamma \vdash A$ *in classical logic, then there is* $\pi' : \left\{(\gamma)^N : \gamma \in \Gamma\right\} \vdash (A)^N$ *in intuitionistic logic.*

We will not prove this theorem: who is interested can inspect it having a look at the references at the end of this lesson.

The proposition has a number of consequences: the relevant ones to us are

- each classical theory and, thus, each classical proof can be translated into intuitionistic logic, yielding a classically equivalent result. So, classical logic is not really more expressive than intuitionistic logic.
- Intuitionistic logic is more expressive than classical logic since it allows to distinguish formulae which are classically equivalent.

References

A good introduction to the constructive way of reasoning can be found in *A.S. Troelstra* and *D. van Dalen*, Constructivism in Mathematics, volume I, Studies in Logic and the Foundations of Mathematics 121, Elsevier, (1988), ISBN 0-444-70506-6.

There are many ways to translate intuitionistic logic into classical logic. A survey can be found in *A.S. Troelstra* and *H. Schwichtenberg*, Basic Proof Theory, Cambridge Tracts in Theoretical Computer Science 43, Cambridge: Cambridge University Press, (1996).

Mathematical Logic

Lecture 16

Dr Marco Benini

marco.benini@uninsubria.it

Dipartimento di Scienza e Alta Tecnologia
Università degli Studi dell'Insubria

a.a. 2016/17

Syllabus

Constructive mathematics:

- Heyting algebra
- Semantics
- Soundness
- Completeness
- Variations on the theme

Heyting algebra

Definition 16.1 (Heyting algebra)

A *Heyting algebra* $\mathscr{H} = \langle H; \leq \rangle$ is a bounded lattice such that, for every $x, y \in H$, there is $c \in H$, the *relative pseudo-complement* of x with respect to y, notation $x \supset y$, such that

1. $x \wedge c \leq y$;
2. for every $z \in H$ such that $x \wedge z \leq y$, $z \leq c$.

The relative pseudo-complement of $x \in H$ with respect to $\bot$ is called the *pseudo-complement* of x and it is denoted by $\neg x$.

Heyting algebra

Examples:

- Every Boolean algebra is also a Heyting algebra.
- Every totally ordered set forming a bounded lattice is a Heyting algebra. In particular, $x \supset y = y$ when $y < x$, and $x \supset y = \top$ otherwise.
- The lattice of open sets in any topology is a Heyting algebra. In particular, $A \supset B$ is the interior of $A^c \cup B$.

The last example shows that a Heyting algebra is not always a Boolean algebra, since the interior of $A^c \cup B$ is usually different from $A^c \cup B$, or, in logical terms, $A \supset B \neq \neg A \vee B$.

Heyting algebra

Fact 16.2

In any Heyting algebra, for each element x, $x \wedge \neg x = \bot$.

Proof.

By definition of bottom and pseudo-complement, $\bot \leq x \wedge \neg x \leq \bot$. □

Fact 16.3

In any Heyting algebra, for all elements x and y, $x \leq y$ if and only if $x \supset y = \top$.

Proof.

Since $x = x \wedge \top$, if $x \leq y$, $x \supset y = \top$ being $\top$ the maximal element z such that $x \wedge z \leq y$. Conversely, if $x \supset y = \top$, then $x \wedge (x \supset y) = x \wedge \top = x \leq y$ by definition of pseudo-complement. □

Heyting algebra

Fact 16.4

There is a Heyting algebra such that, for some element x, $x \vee \neg x \neq \top$.

Proof.

Consider the total order $0 < 1/2 < 1$. It is immediate to check that it is a Heyting algebra. But $1/2 \vee \neg 1/2 = 1/2 \vee 0 = 1/2 \neq 1 = \top$. □

Proposition 16.5

Every Heyting algebra is a distributive lattice.

Proof.

It suffices to prove $x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$. By definition of $\vee$, $y \leq y \vee z$ and $z \leq y \vee z$, thus, by definition of $\wedge$, $x \wedge y \leq x$ and $x \wedge y \leq y \leq y \vee z$, so $x \wedge y \leq x \wedge (y \vee z)$. Symmetrically, it holds that $x \wedge z \leq x \wedge (y \vee z)$. Then, by definition of $\vee$, $(x \wedge y) \vee (x \wedge z) \leq x \wedge (y \vee z)$.
Conversely, $x \wedge y \leq (x \wedge y) \vee (x \wedge z)$ and $x \wedge z \leq (x \wedge y) \vee (x \wedge z)$ by definition of $\vee$. So, $y \leq (x \supset (x \wedge y) \vee (x \wedge z))$ and $z \leq (x \supset (x \wedge y) \vee (x \wedge z))$ by definition of $\supset$, thus, by definition of $\vee$, $y \vee z \leq (x \supset (x \wedge y) \vee (x \wedge z))$. Then, by definition of $\supset$, $x \wedge (y \vee z) \leq (x \wedge y) \vee (x \wedge z)$. □

Propositional semantics

For the sake of simplicity, we will consider just the pure logic instead of a generic theory in the following. The results can be naturally generalised.

Definition 16.6 (Semantics)

Fixed a Heyting algebra $\mathscr{H} = \langle H; \leq \rangle$ and a map $\nu \colon V \to H$, evaluating each variable in some element of $\mathscr{H}$, the meaning $[\![A]\!]$ of a propositional formula A is a map from the set of formulae to H, inductively defined as

1. if $A \equiv x$, a variable, $[\![A]\!] = \nu(x)$;
2. $[\![\top]\!] = \top$ and $[\![\bot]\!] = \bot$;
3. $[\![B \wedge C]\!] = [\![B]\!] \wedge [\![C]\!]$, $[\![B \vee C]\!] = [\![B]\!] \vee [\![C]\!]$, $[\![B \supset C]\!] = [\![B]\!] \supset [\![C]\!]$, and $[\![\neg B]\!] = \neg [\![B]\!]$.

We say that a formula A is *valid* or *true* in the *model* $(\mathscr{H}, \nu)$ when $[\![A]\!] = \top$.

Soundness

Theorem 16.7 (Soundness)

If $\pi : \Gamma \vdash A$ is a proof in the intuitionistic natural deduction calculus, then, in every model $(\mathscr{H}, \nu)$ such that each $G \in \Gamma$ is valid, A is true.

Proof. (i)

Fixed a generic model, by induction on the structure of a proof $\pi : \Delta \vdash B$, with Δ a finite set of assumptions, we prove that $\bigwedge_{D \in \Delta} [\![D]\!] \leq [\![B]\!]$:

- if π is a proof by assumption, $B \in \Delta$, so $\bigwedge_{D \in \Delta} [\![D]\!] \leq [\![B]\!]$ by definition of $\wedge$.
- if π is an instance of $\top$-introduction, $B \equiv \top$, thus, by definition of $\top$, $\bigwedge_{D \in \Delta} [\![D]\!] \leq \top = [\![B]\!]$.
- if π is an instance of $\bot$-elimination, by induction hypothesis and by definition of $\bot$, $\bigwedge_{D \in \Delta} [\![D]\!] \leq [\![\bot]\!] = \bot \leq [\![B]\!]$.
- if π is an instance of $\wedge$-introduction, $B \equiv B_1 \wedge B_2$ and, by induction hypothesis, $\bigwedge_{D \in \Delta} [\![D]\!] \leq [\![B_1]\!]$ and $\bigwedge_{D \in \Delta} [\![D]\!] \leq [\![B_2]\!]$, so, by definition of $\wedge$, $\bigwedge_{D \in \Delta} [\![D]\!] \leq [\![B_1]\!] \wedge [\![B_2]\!] = [\![B_1 \wedge B_2]\!] = [\![B]\!]$. ↪

Soundness

↪ Proof. (ii)

- if π is an instance of $\wedge_1$-elimination or $\wedge_2$-elimination, then, by induction hypothesis, $\bigwedge_{D\in\Delta}[\![D]\!] \leq [\![B \wedge B_1]\!] = [\![B]\!] \wedge [\![B_1]\!]$ or $\bigwedge_{D\in\Delta}[\![D]\!] \leq [\![B_1 \wedge B]\!] = = [\![B_1]\!] \wedge [\![B]\!]$, respectively. Thus, by definition of $\wedge$, $\bigwedge_{D\in\Delta}[\![D]\!] \leq [\![B]\!]$ in both cases.
- if π is an instance of $\vee_1$-introduction or $\vee_2$-introduction, then $B \equiv B_1 \vee B_2$ and, by induction hypothesis, $\bigwedge_{D\in\Delta}[\![D]\!] \leq [\![B_1]\!]$ or $\bigwedge_{D\in\Delta}[\![D]\!] \leq [\![B_2]\!]$, respectively. Thus, by definition of $\vee$, $\bigwedge_{D\in\Delta}[\![D]\!] \leq [\![B_1]\!] \vee [\![B_2]\!] = [\![B_1 \vee B_2]\!] = [\![B]\!]$ in both cases.
- if π is an instance of $\vee$-elimination, by induction hypothesis, $[\![C_1]\!] \wedge \bigwedge_{D\in\Delta}[\![D]\!] \leq [\![B]\!]$ and $[\![C_2]\!] \wedge \bigwedge_{D\in\Delta}[\![D]\!] \leq [\![B]\!]$, so, by definition of $\supset$, $[\![C_1]\!] \leq \bigwedge_{D\in\Delta}[\![D]\!] \supset [\![B]\!]$ and $[\![C_2]\!] \leq \bigwedge_{D\in\Delta}[\![D]\!] \supset [\![B]\!]$, thus $[\![C_1]\!] \vee [\![C_2]\!] = [\![C_1 \vee C_2]\!] \leq \bigwedge_{D\in\Delta}[\![D]\!] \supset [\![B]\!]$. Hence, by definition of $\supset$, $[\![C_1 \vee C_2]\!] \wedge \bigwedge_{D\in\Delta}[\![D]\!] \leq [\![B]\!]$.
  Since, by induction hypothesis, $\bigwedge_{D\in\Delta}[\![D]\!] \leq [\![C_1 \vee C_2]\!]$, by definition of $\wedge$, $[\![C_1 \vee C_2]\!] \wedge \bigwedge_{D\in\Delta}[\![D]\!] = \bigwedge_{D\in\Delta}[\![D]\!] \leq [\![B]\!]$. ↪

Soundness

↪ Proof. (iii)

- if π is an instance of $\supset$-introduction, $B \equiv B_1 \supset B_2$ and, by induction hypothesis, $[\![B_1]\!] \wedge \bigwedge_{D \in \Delta} [\![D]\!] \leq [\![B_2]\!]$. So, by definition of $\supset$, $\bigwedge_{D \in \Delta} [\![D]\!] \leq [\![B_1]\!] \supset [\![B_2]\!] = [\![B_1 \supset B_2]\!] = [\![B]\!]$.
- if π is an instance of $\supset$-elimination, by induction hypothesis, $\bigwedge_{D \in \Delta} [\![D]\!] \leq [\![C \supset B]\!] = [\![C]\!] \supset [\![B]\!]$ thus, by definition of $\supset$, $[\![C]\!] \wedge \bigwedge_{D \in \Delta} [\![D]\!] \leq [\![B]\!]$. Since, by induction hypothesis, $\bigwedge_{D \in \Delta} [\![D]\!] \leq [\![C]\!]$, by definition of $\wedge$, $[\![C]\!] \wedge \bigwedge_{D \in \Delta} [\![D]\!] = \bigwedge_{D \in \Delta} [\![D]\!] \leq [\![B]\!]$.
- if π is an instance of $\neg$-introduction, $B \equiv \neg C$ and, by induction hypothesis, $[\![C]\!] \wedge \bigwedge_{D \in \Delta} [\![D]\!] \leq [\![\bot]\!] = \bot$. So, by definition of $\neg$, $\bigwedge_{D \in \Delta} [\![D]\!] \leq \neg [\![C]\!] = [\![\neg C]\!] = [\![B]\!]$.
- if π is an instance of $\neg$-elimination, by induction hypothesis, $\bigwedge_{D \in \Delta} [\![D]\!] \leq [\![\neg C]\!] = \neg [\![C]\!]$ thus, by definition of $\neg$, $[\![C]\!] \wedge \bigwedge_{D \in \Delta} [\![D]\!] \leq [\![\bot]\!]$. Since, by induction hypothesis, $\bigwedge_{D \in \Delta} [\![D]\!] \leq [\![C]\!]$, by definition of $\wedge$, $[\![C]\!] \wedge \bigwedge_{D \in \Delta} [\![D]\!] = \bigwedge_{D \in \Delta} [\![D]\!] \leq [\![\bot]\!] = \bot \leq [\![B]\!]$, by definition of $\bot$. ↪

Soundness

↪ Proof. (iv)

Now, consider $\pi : \Gamma \vdash A$ as in the statement of the theorem: since the proof π uses just a finite number of assumptions $\Gamma_0 \subseteq \Gamma$, by the induction above, $\bigwedge_{G \in \Gamma_0} [\![G]\!] \leq [\![A]\!]$. But, for each $G \in \Gamma$, $[\![G]\!] = \top$ by hypothesis, thus $\bigwedge_{G \in \Gamma_0} [\![G]\!] = \top \leq [\![A]\!] \leq \top$, by definition of $\top$. So, by anti-symmetry of $\leq$, $[\![A]\!] = \top$. □

Completeness

We will show a simplified completeness result. A more general result can be easily obtained by extending the presented core along the guidelines we followed in the classical case.

Theorem 16.8 (Completeness)

If the propositional formula A is valid in any Heyting model $(\mathscr{H}; v)$, then A is provable in the propositional natural deduction calculus for intuitionistic logic.

Proof. (i)

Let F be the collection of all formulae. We define $A \sim B$ if and only $\vdash A = B$. Evidently, $\sim$ is an equivalence relation over F:

- $A \sim A$ since $\vdash A \supset A$;
- if $A \sim B$ then $\vdash A \supset B$ and $\vdash B \supset A$, so $B \sim A$;
- if $A \sim B$ and $B \sim C$ then $\vdash A \supset B$ and $\vdash B \supset C$, thus $\vdash A \supset C$, but also $\vdash C \supset B$ and $\vdash B \supset A$, so $\vdash C \supset A$, thus $A \sim C$. ↪

Completeness

↪ Proof. (ii)

Let $H = F/\sim$ and let $[A]_\sim \leq [B]_\sim$ exactly when $A \vdash B$. Then $\langle H; \leq \rangle$ is an order since

- $[A]_\sim \leq [A]_\sim$ because $A \vdash A$;
- if $[A]_\sim \leq [B]_\sim$ and $[B]_\sim \leq [A]_\sim$, then $A \vdash B$ and $B \vdash A$, so $\vdash A = B$, that is $A \sim B$, i.e., $[A]_\sim = [B]_\sim$;
- if $[A]_\sim \leq [B]_\sim$ and $[B]_\sim \leq [C]_\sim$, then $A \vdash B$ and $B \vdash C$, so $A \vdash C$, that is, $[A]_\sim \leq [C]_\sim$.

Also, $\langle H; \leq \rangle$ is bounded:

- $\bot = [\bot]_\sim$, in fact, $\bot \vdash A$ for any formula A by $\bot$-elimination, so $[\bot]_\sim \leq [A]_\sim$;
- $\top = [\top]_\sim$, in fact, $A \vdash \top$ for any formula A by $\top$-introduction, so $[A]_\sim \leq [\top]_\sim$. ↪

Completeness

↪ Proof. (iii)

Moreover, $\langle H; \leq \rangle$ is a lattice:

- $[A]_\sim \wedge [B]_\sim = [A \wedge B]_\sim$, in fact, $A \wedge B \vdash A$ and $A \wedge B \vdash B$ by $\wedge$-elimination, so $[A \wedge B]_\sim \leq [A]_\sim$ and $[A \wedge B]_\sim \leq [B]_\sim$; if $[C]_\sim \leq [A]_\sim$ and $[C]_\sim \leq [B]_\sim$, then $C \vdash A$ and $C \vdash B$, so $C \vdash A \wedge B$ by $\wedge$-introduction, that is, $[C]_\sim \leq [A \wedge B]_\sim$;
- $[A]_\sim \vee [B]_\sim = [A \vee B]_\sim$, in fact, $A \vdash A \vee B$ and $B \vdash A \vee B$ by $\vee$-introduction, so $[A]_\sim \leq [A \vee B]_\sim$ and $[B]_\sim \leq [A \vee B]_\sim$; if $[A]_\sim \leq [C]_\sim$ and $[B]_\sim \leq [C]_\sim$, then $A \vdash C$ and $B \vdash C$, so $A \vee B \vdash C$ by $\vee$-elimination, that is, $[A \vee B]_\sim \leq [C]_\sim$.

Finally, $\langle H; \leq \rangle$ is a Heyting algebra: $[A]_\sim \supset [B]_\sim = [A \supset B]_\sim$, in fact, $A \wedge (A \supset B) \vdash B$ by $\supset$-elimination, so $[A \wedge (A \supset B)]_\sim = [A]_\sim \wedge [A \supset B]_\sim \leq [B]_\sim$; when $[A]_\sim \wedge [C]_\sim = [A \wedge C]_\sim \leq [B]_\sim$, $A \wedge C \vdash B$, so $C \vdash A \supset B$ by $\supset$-introduction , that is $[C]_\sim \leq [A \supset B]_\sim$. It is worth noticing that $\neg [A]_\sim = [\neg A]_\sim$ since $\vdash \neg A = (A \supset \bot)$. ↪

Completeness

↪ Proof. (iv)

Let $\nu\colon V \to H$ be $\nu(x) = [x]_\sim$ for any variable x.

By induction on the structure of A, we prove that $[\![A]\!] = [A]_\sim$ in $((H;\leq),\nu)$:

- if $A \equiv x$, a variable, by definition $[\![A]\!] = \nu(x) = [x]_\sim = [A]_\sim$;
- if $A \equiv \top$, $[\![A]\!] = \top = [\top]_\sim$;
- if $A \equiv \bot$, $[\![A]\!] = \bot = [\bot]_\sim$;
- if $A \equiv B \wedge C$, by induction hypothesis,
  $[\![A]\!] = [\![B]\!] \wedge [\![C]\!] = [B]_\sim \wedge [C]_\sim = [B \wedge C]_\sim = [A]_\sim$;
- if $A \equiv B \vee C$, by induction hypothesis,
  $[\![A]\!] = [\![B]\!] \vee [\![C]\!] = [B]_\sim \vee [C]_\sim = [B \vee C]_\sim = [A]_\sim$;
- if $A \equiv B \supset C$, by induction hypothesis,
  $[\![A]\!] = [\![B]\!] \supset [\![C]\!] = [B]_\sim \supset [C]_\sim = [B \supset C]_\sim = [A]_\sim$;
- if $A \equiv \neg B$, by induction hypothesis, $[\![A]\!] = \neg[\![B]\!] = \neg[B]_\sim = [\neg B]_\sim = [A]_\sim$. ↪

Completeness

$\hookrightarrow$ Proof. (v)

By hypothesis of the theorem, A is valid in any model, that is $[\![A]\!] = \top$ in any model, so, in particular $[\![A]\!] = \top$ in $((H;\leq),\nu)$. But in $((H;\leq),\nu)$, $[A]_\sim = [\![A]\!] = \top = [\top]_\sim$, thus $A \sim \top$, that is $\vdash A \supset \top$ and $\vdash \top \supset A$. By $\top$-introduction and $\vdash \top \supset A$, we get that $\vdash A$. $\square$

Variations on the theme

The algebraic semantics based on Heyting algebras can be generalised to provide a meaning to first-order intuitionistic logic.

There are many ways to achieve this result, obtaining a soundness and completeness theorem:

- Heyting categories;
- Kripke semantics;
- logical categories.

Variations on the theme

Heyting categories are categories with a somewhat involved structure such that the class of sub-objects of any object form a Heyting category, ordered by the factorisation of morphisms.

Although it is beyond the scope of these lessons to provide a formal account, the idea is that quantifiers get a meaning by considering the maximal and the minimal element in a Heyting algebra which is related to the algebra used to interpret the quantified formula, so that these extreme elements are generated by the relation of algebras, which models the elimination of the quantified variable.

Variations on the theme

Since any topos is also a Heyting category, one can limit the class of models to toposes. It turns out that it suffices to prove a completeness result.

Moreover, a further limitation to Grothendieck toposes suffices, too. This becomes interesting because a topos of sheaves, the prototypical Grotendieck topos, provides a model which is composed by a collection of almost classical models, a la Tarski, but in the internal set theory of the topos, linked together by a relation modelling the growth of knowledge implicit in the constructive nature of intuitionistic first-order logic.

These models suffice to prove a completeness result, and their classical set-theoretic version is known as *Kripke semantics*, and it is usually built up from the usual set theory.

Variations on the theme

On a different line, by using categories naturally equipped with a Heyting algebra and a sort of topological structure, modelling the link between a quantified formula and its instances through the introduction and elimination inference rules, one obtains another sound and complete semantics.

These categories are known as *logical categories*.

All these semantics are strictly related one to the other, emphasising some aspects of the deep nature of constructive logical systems, and this is the reason why all of them have been developed.

References

Heyting algebras have been introduced by Arend Heyting in 1930 to formalise intuitionistic logic. An algebraic introduction to Heyting algebras is in *George Grätzer*, General Lattice Theory, second edition, Birkhäuser, (1996), ISBN 978-3-7643-6996-5.

The soundness theorem as presented, is folklore: the actual presentation derives from the generalised result on the internal logic of topos theory, which is based on the fact that the lattice of subobjects of the terminal object in a topos forms a Heyting algebra. For those interested the details can be found in *R. Goldblatt*, Topoi: The Categorical Analysis of Logic, Dover Publishing, (2006), ISBN 0-486-45026-0.

The proof of the completeness theorem has been adapted from the categorical version in *P. Johnstone*, Sketches of an Elephant: A Topos Theory Compendium, two volumes, Oxford University Press (2002), ISBN 978-0-19-853425-9 and 978-0-19-851598-2.

References

Heyting categories are defined in the same text. The internal logic of a topos has been introduced by W. Lawvere, and an approachable text is *R. Goldblatt*, Topoi: The Categorical Analysis of Logic, Dover Publishing, (2006), ISBN 0-486-45026-0.

On the contrary, logic categories have been introduced by M. Benini, and a quite technical survey can be found in *M. Benini*, Proof-Oriented Categorical Semantics, in D. Probst, P. Schuster eds., Concepts of Proof in Mathematics, Philosophy, and Computer Science, Ontos Mathematical Logic 6, De Gruyter, pp. 41-68 (2016).

Mathematical Logic

Lecture 17

Dr Marco Benini

marco.benini@uninsubria.it

Dipartimento di Scienza e Alta Tecnologia
Università degli Studi dell'Insubria

a.a. 2016/17

Syllabus

Constructive mathematics:

- λ calculus
- The simple theory of types

λ-calculus

The λ-calculus is a family of formal systems, based on Alonzo Church's work in the 1930s. These systems are deputed to describe computable functions using the simplest syntax. Surprisingly, not only they describe computable functions, but, when equipped with types, they show a hidden link between logic and computability.

In this lecture, we want to introduce the λ-calculus and its simplest typed version. Our aim is to illustrate the general aspects of the theory and to derive a few results, the one we will use in the following lessons.

In many cases, we will avoid proving the results we will introduce. This is done on purpose: the simplicity of the formal system has as a natural counterpart a deep and complex technical development. Although this technical part has many pearls, which shed light to some important aspects of computability, it lies beyond the aims of this course.

λ-calculus

Definition 17.1 (λ-term)

Fixed a set V of *variables*, which is both infinite and recursive, a *λ-term* is inductively defined as:

- any $x \in V$ is a λ-term, and $\mathrm{FV}(x) = \{x\}$;
- if M and N are λ-terms, so is $(M \cdot N)$, called *application*, and $\mathrm{FV}(MN) = \mathrm{FV}(M) \cup \mathrm{FV}(N)$;
- if $x \in V$ and M is a λ-term, so is $(\lambda x.\, M)$, called *abstraction*, and $\mathrm{FV}(\lambda x.\, M) = \mathrm{FV}(M) \setminus \{x\}$.

The set $\mathrm{FV}(M)$ is called the set of *free variables* in M, and the variables in M not occurring in $\mathrm{FV}(M)$ are said to be *bound*.

Example 17.2

$(\lambda x.x)$ is a λ-term with no free variables, representing the identity function.

λ-calculus

As usual, to simplify notation, we introduce a number of conventions:

- outermost parentheses are not written: $\lambda x.x$ instead of $(\lambda x.x)$;
- a sequence of consecutive abstraction is grouped: $\lambda x,y.x\cdot y$ instead of $\lambda x.(\lambda y.x\cdot y)$;
- we treat application as a product, omitting the dot: xy instead of $x\cdot y$;
- we assume application associates to the left: xyz instead of $(xy)z$.

Also, we use the term *combinator* to denote a λ-term having no free variables.

Example 17.3

The following are combinators

- $I \equiv \lambda x.x$;
- $K \equiv \lambda x,y.x$;
- $S \equiv \lambda x,y,z.(xy)(xz)$;
- $\Omega \equiv (\lambda x.xx)(\lambda x.xx)$.

Definition 17.4 (Substitution)

For any M, N λ-terms, and x variable, $M[N/x]$ is the *substitution* of x with N in M, defined by induction on M as:

- $x[N/x] \equiv N$;
- $y[N/x] \equiv y$, when $x \not\equiv y$;
- $(PQ)[N/x] \equiv (P[N/x])(Q[N/x])$;
- $(\lambda x.P)[N/x] \equiv \lambda x.P$;
- $(\lambda y.P)[N/x] \equiv \lambda y.P[N/x]$, when $x \not\equiv y$ and $y \notin \mathrm{FV}(N)$;
- $(\lambda y.P)[N/x] \equiv \lambda z.(P[z/y])[N/x]$, when $x \not\equiv y$ and $y \in \mathrm{FV}(N)$ and $z \notin \mathrm{FV}(P) \cup \mathrm{FV}(N)$.

In the last clause, the z variable is said to be *new*, and it is always possible to choose a z which satisfies the constraint.

The purpose of the last clause is to prevent variable capturing.

λ-calculus

Definition 17.5 (α-equivalence)

The λ-terms M and N are α-equivalent, $M =_\alpha N$, when

- $M \equiv N$;
- $M \equiv PQ$, $N \equiv P'Q'$, and $P =_\alpha P'$ and $Q =_\alpha Q'$;
- $M \equiv \lambda x.P$, $N \equiv \lambda x.P'$, and $P =_\alpha P'$;
- $M \equiv \lambda x.P$ and $N \equiv \lambda y.P[y/x]$

So, two λ-terms are α-equivalent when they differ for the names of bound variables only.

It is immediate to see that α-equivalence is an equivalence relation, but it is also a *congruence* with respect to substitution:

Proposition 17.6

If $M =_\alpha M'$ *and* $N =_\alpha N'$, *then* $M[N/x] =_\alpha M'[N'/x]$.

λ-calculus

Definition 17.7 (β-reduction)

The binary relation between λ-terms $M \rhd_{1,\beta} N$, *M β-reduces to N in one step*, holds if and only if $M \equiv M'[(\lambda x.P) \cdot Q/z]$ and $N \equiv N'[(P[Q/x])/z]$. We say that *M β-reduces* to N, $M \rhd_\beta N$, when there is a finite sequence $P_1, \ldots, P_n$ such that $M \equiv P_1$, $N \equiv P_n$ and, for each $1 \leq i < n$, $P_i \rhd_{1,\beta} P_{i+1}$.

In the λ-calculus, computation is performed by β-reduction.

Definition 17.8 (β-normal form)

A term N is said to be in *β-normal form* when it does not contain any subterm of the form $(\lambda x.P)Q$.

With respect to computations, λ-terms in β-normal form represent the values.

λ-calculus

Theorem 17.9 (Church-Rosser)

If $M \rhd_\beta P$ and $M \rhd_\beta Q$, then there is a λ-term R such that $P \rhd_\beta R$ and $Q \rhd_\beta R$.

Corollary 17.10

If $M \rhd_\beta N$ and N is a β-normal form, then N is unique up to α-equivalence.

The Church-Rosser Theorem and its corollary say that, although the computation in λ-calculus is non-deterministic, the result, when it exists, is uniquely determined.

Definition 17.11 (β-equality)

We say that P is β-equivalent to Q, $P =_\beta Q$, when there is a finite sequence $R_1, \ldots, R_n$ such that $P \equiv R_1$, $Q \equiv R_n$, and, for all $1 \leq i < n$, $R_i \rhd_{1,\beta} R_{i+1}$, or $R_{i+1} \rhd_{1,\beta} R_i$, or $R_i =_\alpha R_{i+1}$.

Theorem 17.12 (Fixed point)

There is a combinator $\mathbf{Y}$ *such that* $\mathbf{Y}x =_\beta x(\mathbf{Y}x)$.

Proof.

Let $U \equiv \lambda u, x.x(uux)$, and let $\mathbf{Y} \equiv UU$. Then
$\mathbf{Y}x \equiv (\lambda u, x.x(uux))Ux \rhd_\beta (\lambda x.x(UUx))x \rhd_\beta x(UUx) \equiv x(\mathbf{Y}x)$. □

The proof of the fixed point theorem as above, is due to Alan Turing.

The fixed point theorem says that, every λ-term, when thought of as a function, has a fixed point which is calculated by the $\mathbf{Y}$ combinator. This is an important property which suggests that each function which can be represented as a λ-term, has to be continuous.

Representable functions

Definition 17.13 (Numerals)

For every $n \in \mathbb{N}$, the *Church numeral* $\overline{n}$ is a λ-term inductively defined as:

- $\overline{0} = \lambda x, y.\, y$;
- $\overline{n+1} = \lambda x, y.\, x(\overline{n}xy)$.

Definition 17.14 (Representable functions)

Let $f : \mathbb{N}^k \to \mathbb{N}$ be a partial function. A λ-term F is said to *represent* the function f when

- for all $n_1, \ldots, n_k \in \mathbb{N}$, if $f(n_1, \ldots, n_k) = m$, then $F\overline{n_1}, \ldots, \overline{n_k} = \overline{m}$;
- for all $n_1, \ldots, n_k \in \mathbb{N}$, if $f(n_1, \ldots, n_k)$ is undefined, then $F\overline{n_1}, \ldots, \overline{n_k}$ has no β-normal form.

Theorem 17.15

Every partial recursive function can be represented in the λ-calculus.

Representable functions

The proof of the theorem is difficult beyond the aim of this course. But we will show a few examples to justify it.

Example 17.16

The successor function is represented by $\lambda x, s, z.\, s(xsz)$.
Addition is represented by $\lambda x, y, s, z.\, xs(ysz)$.
Multiplication is represented by $\lambda x, y, s.\, x(ys)$.
Exponentiation is represented by $\lambda x, y.\, yx$

Example 17.17

The Boolean values $\top$ and $\bot$ are represented as $\lambda x, y.\, y$ and $\lambda x, y.\, x$, respectively.
Then, 'if x then y else z' is represented by $\lambda x, y, z.\, xzy$.
In fact, if $\overline{\bot}$ then A else $B \equiv (\lambda x, y, z.\, xzy)(\lambda x, y.\, x)AB =_\beta (\lambda y, z.\, (\lambda x, y.\, x)zy)AB =_\beta (\lambda y, z.\, z)AB =_\beta B$, while if $\overline{\top}$ then A else $B \equiv (\lambda x, y, z.\, xzy)(\lambda x, y.\, y)AB =_\beta (\lambda y, z.\, (\lambda x, y.\, y)zy)AB =_\beta (\lambda y, z.\, y)AB =_\beta A$.

Representable functions

To get a clue why these representations work, we could read them as computations over logical structures. For example, natural numbers are inductively defined from 0 and the successor. Hence, a model for the naturals is specified when we provide a set together with a way to interpret 0 as some specific element, and the successor as an injective function which transforms an element into another.

Consider $\overline{0} \equiv \lambda x, y.\, y$: this is a function from the model which provides an element of the model. The model is specified by providing the specification of the successor and the specification of zero. The result is the specification of 0.

Consider $\overline{n+1} \equiv \lambda x, y.\, x(\overline{n}xy)$: since $\overline{n}$ transforms a model into a number, the term $\overline{n}xy$ evaluates to n in the model (x, y). But x stands for the successor function, so we are taking the successor of n in the model.

Thus, $x + y$ is calculated by interpreting x in a model where the successor function is given, but the zero element is ysz, i.e., the number which stands for y in the model.

Similarly, xy is calculated by interpreting x in a model where the successor function moves by y steps at once.

Simple theory of types

Definition 17.18 (Type)

Fixed a denumerable set V_T of *type variables*, a *type* is inductively defined as follows:

- $x \in V_T$ is a type;
- 0 and 1 are types;
- if α and β are types, so are $(\alpha \times \beta)$, $(\alpha + \beta)$, and $(\alpha \to \beta)$.

As usual, we omit parentheses when they are not strictly needed: $\times$ binds stronger that $+$, and $+$ binds stronger than $\to$, so $\alpha \times \beta + \gamma \to (\alpha + \gamma) \times (\beta + \gamma)$ stands for $((\alpha \times \beta) + \gamma) \to ((\alpha + \gamma) \times (\beta + \gamma))$.

A type is used to constrain the main entity of interest in the theory of types, the *term*.

Simple theory of types

Definition 17.19 (Term)

Fixed a family $\{V_\alpha\}_\alpha$ of *variables*, indexed by the collection of types, such that, for each α, V_α is denumerable and distinct from the set of type variables, and such that $V_\alpha \cap V_\beta = \emptyset$ whenever $\alpha \neq \beta$, a *term* $t : \alpha$ of type α, along with the set of its *free variables*, is inductively defined as:

- if $x \in V_\alpha$ for some type α, $x : \alpha$ is a term, and $\mathrm{FV}(x : \alpha) = \{x : \alpha\}$;
- $* : 1$ is a term and $\mathrm{FV}(* : 1) = \emptyset$;
- for each type α, $\Box_\alpha : 0 \to \alpha$ is a term and $FV(\Box_\alpha : 0 \to \alpha) = \emptyset$;
- if $A : \alpha$ and $B : \beta$ are terms, $\langle A, B\rangle : \alpha \times \beta$ is a term and $\mathrm{FV}(\langle A, B\rangle : \alpha \times \beta) = \mathrm{FV}(A : \alpha) \cup \mathrm{FV}(B : \beta)$;
- if $A : \alpha \times \beta$ is a term, so are $\pi_1 A : \alpha$ and $\pi_2 A : \beta$, and $\mathrm{FV}(\pi_1 A : \alpha) = \mathrm{FV}(\pi_2 A : \beta) = \mathrm{FV}(A : \alpha \times B)$;

↪

Simple theory of types

$\hookrightarrow$ (Term)

- if $A\colon \alpha$ is a term, then, for any type β, $i_1^\beta A\colon \alpha+\beta$ and $i_2^\beta A\colon \beta+\alpha$ are terms and $\mathrm{FV}(i_1^\beta A\colon \alpha+\beta) = \mathrm{FV}(i_2^\beta A\colon \beta+\alpha) = \mathrm{FV}(A\colon \alpha)$;
- if $C\colon \alpha+\beta$, $A\colon \alpha\to\gamma$, and $B\colon \beta\to\gamma$ are terms, so is $\delta(C,A,B)\colon \gamma$, and $\mathrm{FV}(\delta(C,A,B)\colon \gamma) = \mathrm{FV}(C\colon \alpha+\beta)\cup\mathrm{FV}(A\colon \alpha\to\gamma)\cup\mathrm{FV}(B\colon \beta\to\gamma)$;
- if $A\colon \beta$ is a term and $x\in V_\alpha$, then $\lambda x\colon \alpha.A\colon \alpha\to\beta$ is a term and $\mathrm{FV}(\lambda x\colon \alpha.A\colon \alpha\to\beta) = FV(A\colon \beta)\setminus\{x\colon \alpha\}$;
- if $A\colon \alpha$ and $B\colon \alpha\to\beta$ are terms, then $B\cdot A\colon \beta$ is a term and $\mathrm{FV}(B\cdot A\colon \beta) = FV(A\colon \alpha)\cup\mathrm{FV}(B\colon \alpha\to\beta)$.

Terms represent the primitive computational statements.

Simple theory of types

Terms can be *reduced* according to the following rules, where it is assumed that both sides of the equalities are correctly typed:

- $\pi_1\langle A, B\rangle = A$;
- $\pi_2\langle A, B\rangle = B$;
- $\langle \pi_1 A, \pi_2 A\rangle = A$;
- $(\lambda x\colon \alpha. A)\cdot B = A[B/x]$, the act of substituting B for x;
- $\lambda x\colon \alpha.(A\cdot x) = A$, when $x\colon \alpha \notin \mathrm{FV}(A\colon \alpha \to \beta)$;
- $\delta(i_1 C, A, B) = A\cdot C$;
- $\delta(i_2 C, A, B) = B\cdot C$.

It is clear that these rules satisfy the requirements on computable functions.

Simple theory of types

If we restrict to the subsystem whose types are those generated by type variables, $\rightarrow$ and $\times$, and whose terms are, correspondingly, the variables, and those of the form $\lambda x\colon \alpha . A\colon \alpha \rightarrow \beta$, called *abstractions*, $A \cdot B\colon \beta$, called *applications*, $\langle A, B\rangle\colon \alpha \times \beta$, called *pairs*, $\pi_1 A\colon \alpha$ and $\pi_2 A\colon \beta$, called *projections*, we get a subsystem of special interest.

In fact, if we interpret $\times$ as the Cartesian product, and $\rightarrow$ as the function space, we can easily derive a representation of the natural numbers, together with the operations of addition, multiplication and exponentiation, the Boolean values, the if-then-else construction, and so on.
In fact, these representation are nothing but the same we used for the pure, non-typed λ-calculus.

References

A classical, and still excellent introduction to λ-calculus and the simple theory of types is *J.R. Hindley* and *J.P. Seldin*, Lambda-Calculus and Combinators, Cambridge University Press, (2008), ISBN 978-0-521-89885-0.

The link between logical systems, their semantics, and the simple theory of types is illustrated in *P. Johnstone*, Sketches of an Elephant: A Topos Theory Compendium, two volumes, Oxford University Press (2002), ISBN 978-0-19-853425-9 and 978-0-19-851598-2.

The link between λ-calculus, continuous functions, and topological spaces is explained in the fundamental paper *D. Scott*, Continuous lattices, in F.W. Lawvere ed., Toposes, Algebraic Geometry and Logic: Dalhousie University, Halifax, January 16–19, 1971, pp. 97–136, Springer (1972), ISBN 978-3-540-37609-5.

Mathematical Logic

Lecture 18

Dr Marco Benini

marco.benini@uninsubria.it

Dipartimento di Scienza e Alta Tecnologia
Università degli Studi dell'Insubria

a.a. 2016/17

Syllabus

Constructive mathematics:

- Propositions as types
- Proofs, computationally
- Computations, logically
- Variations on the theme

Propositions as types

If we put side by side propositional logical formulae and types in the simple theory of types, we get:

types	formulae
variable	variable
0	$\bot$
1	$\top$
$\alpha \times \beta$	$\alpha \wedge \beta$
$\alpha + \beta$	$\alpha \vee \beta$
$\alpha \to \beta$	$\alpha \supset \beta$

This correspondence shows that we can translate any logical formula in a type and any type in a formula, by a one-to-one map.

Propositions as types

If we put side by side propositional proof in the intuitionistic natural deduction system, and terms in the simple theory of types, we get:

proof	assumption	$\top I$	$\bot E$	$\wedge I$	$\wedge E_{1,2}$	$\vee I_{1,2}$	$\vee E$	$\supset I$	$\supset E$
term	variable	$*$	$\square_\alpha$	$\langle \_, \_ \rangle$	π_1, π_2	i_1^α, i_2^α	δ	λ	$\cdot$

There is an evident one-to-one correspondence, which perfectly matches the one on types.

Propositions as types

Let's examine a couple of examples:

- if $A\colon \alpha$ and $B\colon \beta$ are terms, so is $\langle A, B\rangle\colon \alpha \times \beta$ becomes

$$\dfrac{\begin{array}{cc}\vdots & \vdots \\ A & B \\ \alpha & \beta\end{array}}{\alpha \wedge \beta}\ \wedge\mathsf{I}$$

- if $A\colon \beta$ is a term and $x\colon \alpha$ a variable, then $\lambda x\colon \alpha.A\colon \alpha \to \beta$ becomes

$$\dfrac{\begin{array}{c}[\alpha]^* \\ \vdots \\ A \\ \beta\end{array}}{\alpha \supset \beta}\ \supset\mathsf{I}^*$$

where the label $*$ stands for x.

Propositions as types

The correspondence illustrated so far is known as the *propositions-as-types interpretation*, and also as the *Curry-Howard isomorphism*.

At a first glance, the simple theory of types is just a way to write proofs and formulae as linear expressions instead of adopting the tree-like syntax of natural deduction.

But the logical syntax is coupled with a semantics, and the type theory with a computational meaning, given by the reduction rules.

Computations, logically

Since every formal proof in intuitionistic logic corresponds to a typed term, and typed terms are also λ-terms, each proof is a program which computes something.

It is possible to associate to each proof an object, which is an *evidence* of its type, or its conclusion, if you prefer. So, the evidence of $A \wedge B$ is a pair of evidences for A and B; the evidence of $A \vee B$ is a pair (w, e), with $w \in \{1, 2\}$ telling us which disjunct holds, and e an evidence for it; the evidence of $A \supset B$ is a function mapping any evidence of A into an evidence of B.

These evidences are the intermediate results of the computation performed by the λ-term associated to the proof. So, in a constructive system, proving a statement is, essentially, equivalent to write a computer program satisfying a specification given by the conclusion.

Proofs, computationally

Since typed terms are proofs under the correspondence, we can reduce them to a normal form. Formalising this process leads to state that every proof possesses a normal form.

Thus, considering any proof $\pi : \vdash A \vee B$, it can be reduced to a proof $\pi' : \vdash A \vee B$ in normal form, whose last step is either an instance of $\vee I_1$ or $\vee I_2$. Hence, the conclusion of the last but one step would be either A or B.

Similarly, considering any proof $\pi : \vdash \exists x : s.A$, it can be reduced to a proof $\pi' : \vdash \exists x : s.A$ in normal form, whose last step is an instance of $\exists I$. Hence, the conclusion of the last but one step would be either $A[t/x]$ for some term t, providing a witness to the existential statement.

Variations on the theme

The simple theory of types is just the simplest type theory: many other systems have been analysed, and many of them have a propositions-as-types interpretation, computationally characterising some logical system.

In some cases, like in the constructive type theory, the corresponding logical system is part of the type theory itself. This reflection allows to use such a system to describe mathematical theories, like set theory, inside the type system, becoming part of it. Thus, the type system acts as a *universal* theory, which contains the whole mathematics representable in its logical counterpart.

This way of proceeding has recently lead to a promising approach, which explains computation in terms of algebraic topology (and vice versa). It is called *homotopy type theory*, and it is part of the contemporary frontier of mathematical research. The basic idea is that, by adding a pair of axioms to constructive type theory, one can interpret a computation as a path in some homotopy space. It turns out that paths which are homotopy equivalent can be represented by the same term. Of course, behind this intuition the formal theory is somewhat involved, and still in development...

Normalisation

We want to discuss the normalisation process, which has been sketched above, in the case of intuitionistic propositional logic.

The objective of normalisation is to eliminate the redundant steps in a proof, and to give it a standard format, *minimal*, in a sense.

A natural requirement for a proof in natural deduction is that no conclusion of an introduction rule must be the major premise of an elimination rule. The major premise is the formula containing as principal connective the one which is eliminated by an elimination rule.

Also, another natural requirement is that discharged assumptions should be used in disjunction elimination, while the false elimination rule has to derive a conclusion which is not $\bot$.

Finally, although the previous requirements seems evident, they can be hidden, because of multiple subsequent elimination rules which can be permuted.

Normalisation

The *detour conversions* are deputed to eliminate detours, i.e., redundant elementary steps in a proof given by an introduction rule in the major premise of an elimination rule:

- $\wedge$ rules:

$$\dfrac{\dfrac{\begin{array}{cc}\vdots\,p_1 & \vdots\,p_2\\ A & B\end{array}}{A\wedge B}\wedge\mathrm{I}}{A}\wedge\mathrm{E}_1 \rightsquigarrow \begin{array}{c}\vdots\,p_1\\ A\end{array} \qquad \dfrac{\dfrac{\begin{array}{cc}\vdots\,p_1 & \vdots\,p_2\\ A & B\end{array}}{A\wedge B}\wedge\mathrm{I}}{B}\wedge\mathrm{E}_2 \rightsquigarrow \begin{array}{c}\vdots\,p_2\\ B\end{array}$$

- $\supset$ rules:

$$\dfrac{\dfrac{\begin{array}{c}[A]^*\\ \vdots\,p_1\\ B\end{array}}{A\supset B}\supset\mathrm{I}^* \quad \begin{array}{c}\vdots\,p_2\\ A\end{array}}{B}\supset\mathrm{E} \rightsquigarrow \begin{array}{c}\vdots\,p_2\\ A\\ \vdots\,p_1\\ B\end{array}$$

Normalisation

- $\vee$ rules:

$$\dfrac{\dfrac{\begin{array}{c}\vdots\,p_1\\ A\end{array}}{A \vee B}\vee\mathrm{I}_1 \qquad \begin{array}{c}[A]^*\\ \vdots\,p_2\\ C\end{array} \qquad \begin{array}{c}[B]^*\\ \vdots\,p_3\\ C\end{array}}{C}\vee\mathrm{E}^* \quad \rightsquigarrow \quad \begin{array}{c}\vdots\,p_1\\ A\\ \vdots\,p_2\\ C\end{array}$$

$$\dfrac{\dfrac{\begin{array}{c}\vdots\,p_1\\ B\end{array}}{A \vee B}\vee\mathrm{I}_2 \qquad \begin{array}{c}[A]^1\\ \vdots\,p_2\\ C\end{array} \qquad \begin{array}{c}[B]^1\\ \vdots\,p_3\\ C\end{array}}{C}\vee\mathrm{E}^1 \quad \rightsquigarrow \quad \begin{array}{c}\vdots\,p_1\\ B\\ \vdots\,p_3\\ C\end{array}$$

Normalisation

Since $\neg A \equiv A \supset \bot$, we do not need detour conversions for $\neg$ rules, as soon as we rewrite them as instances of the $\supset$ rules. The conversions for $\supset$ and $\vee$ are justified by Proposition 6.2, which allows to join proofs.

There are no detour conversions for $\bot$ and $\top$, since these connectives lack an introduction and elimination rule, respectively.

It is instructive to see these conversions through the propositions-as-types correspondence: for example, the detour conversion for $\wedge$ becomes $\pi_1\langle p_1, p_2\rangle = p_1$ and $\pi_2\langle p_1, p_2\rangle = p_2$. This observation shows how normalisation in proofs is the same as deriving a normal form for a term in the simple theory of types.

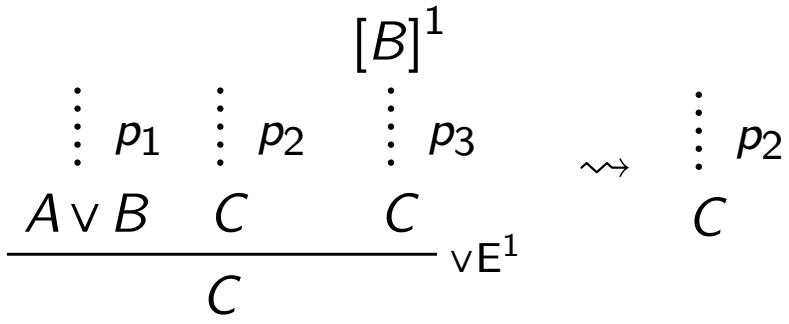
Normalisation

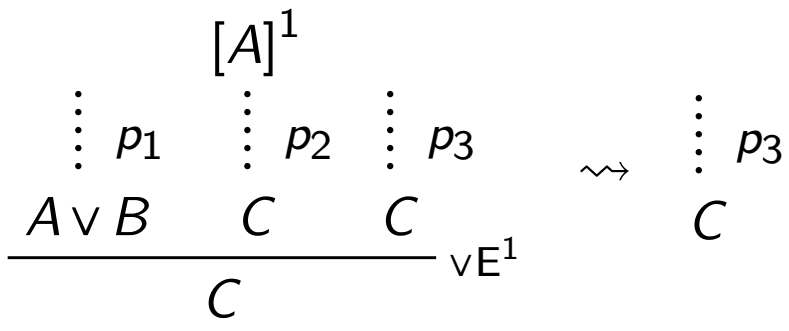
Detour conversions eliminate obviously redundant steps in a proof. However, there are instances of the disjunction elimination rule that are, in fact, redundant, those in which one of the discharged assumptions is not used. This fact leads to define the following *simplification conversions*: if, in

$$\dfrac{\begin{array}{ccc} & [A]^1 & [B]^1 \\ \vdots\, p_1 & \vdots\, p_2 & \vdots\, p_3 \\ A \vee B & C & C \end{array}}{C}\ \vee \mathrm{E}^1$$

either the assumption A in p_2 is not used, or the assumption B in p_3 is not used, then we can use p_2 or p_3, respectively to prove the conclusion.

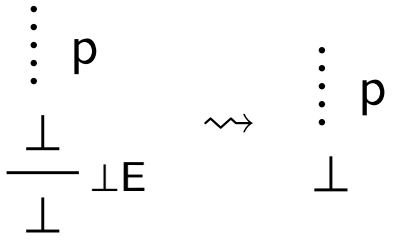
Normalisation





Normalisation

Moreover, the instances of the $\bot$ elimination rule in which the conclusion is $\bot$ are obviously redundant, and we can apply another *simplification conversion* to eliminate them.

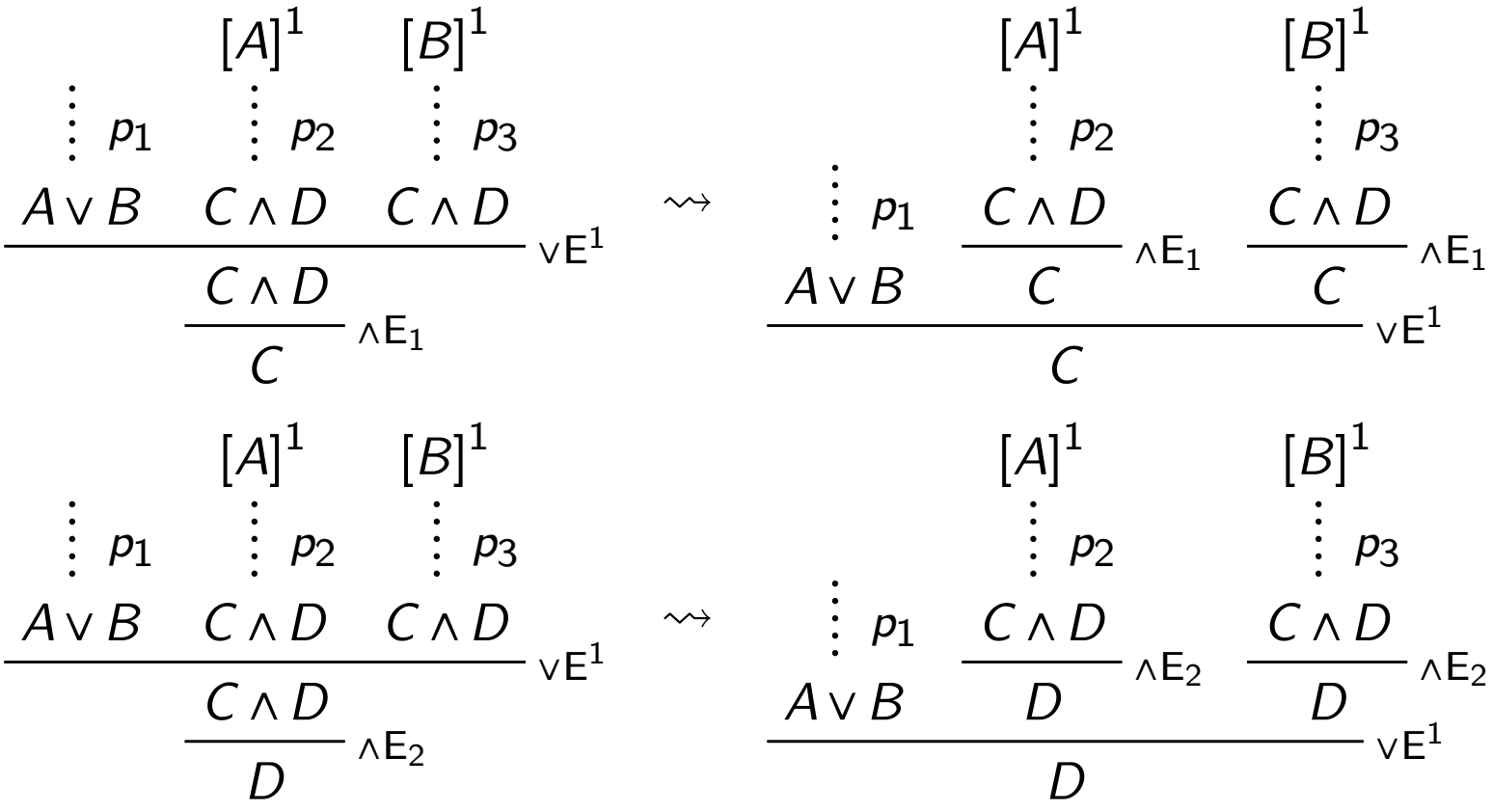


Sometimes, detours and simplifications cannot be directly applied, because they are hidden inside a proof. This happens when we apply an elimination rule whose major premise is an application of the disjunction elimination rule.

In those cases, we can move the disjunction elimination downwards, eventually revealing hidden detours and simplifications. The rules to do so are called *permutation conversions*.

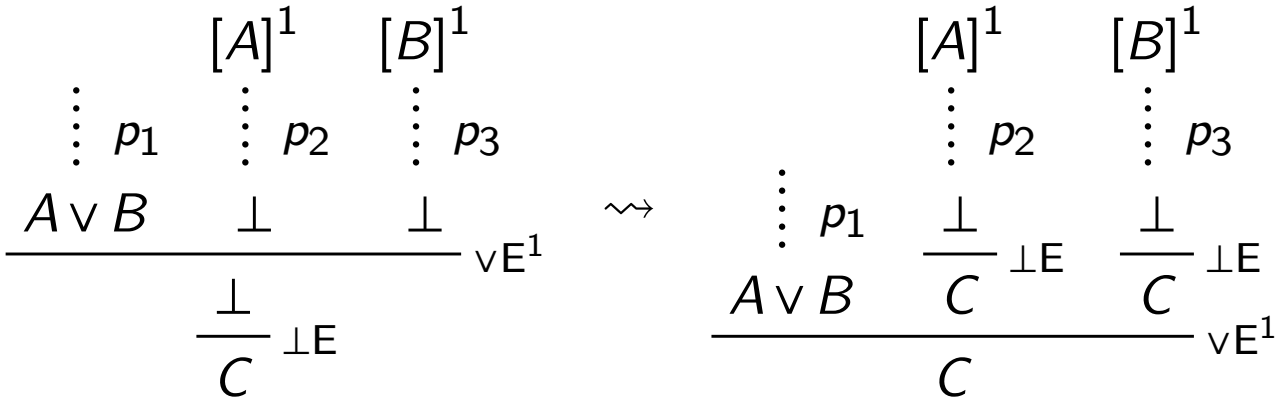
Normalisation

- $\wedge$ elimination:



Normalisation

- $\bot$ elimination:



Normalisation

- $\supset$ elimination:

$$\dfrac{\dfrac{\begin{array}{ccc} & [A]^1 & [B]^1 \\ \vdots\, p_1 & \vdots\, p_2 & \vdots\, p_3 \\ A \vee B & C \supset D & C \supset D \end{array}}{C \supset D}\vee\mathrm{E}^1 \qquad \begin{array}{c}\vdots\, p_4 \\ C\end{array}}{D}\supset\mathrm{E} \quad \rightsquigarrow$$

$$\rightsquigarrow \quad \dfrac{\begin{array}{ccc} \vdots\, p_1 & \dfrac{\begin{array}{cc}[A]^1 & \\ \vdots\, p_2 & \vdots\, p_4 \\ C \supset D & C\end{array}}{D}\supset\mathrm{E} & \dfrac{\begin{array}{cc}[B]^1 & \\ \vdots\, p_3 & \vdots\, p_4 \\ C \supset D & C\end{array}}{D}\supset\mathrm{E} \\ A \vee B & & \end{array}}{D}\vee\mathrm{E}^1$$

Normalisation

- $\vee$ elimination:

$$\dfrac{\begin{array}{ccccc} & [A]^1 & [B]^1 & & \\ \vdots\, p_1 & \vdots\, p_2 & \vdots\, p_3 & [C]^2 & [D]^2 \\ A \vee B & C \vee D & C \vee D & \vdots\, p_4 & \vdots\, p_5 \\ \hline & C \vee D & & E & E \end{array} \vee E^1}{E} \vee E^2$$

$$\rightsquigarrow$$

$$\dfrac{\begin{array}{cccccc} & [A]^1 & [C]^2 & [D]^2 & [B]^1 & [C]^3 & [D]^3 \\ & \vdots\, p_2 & \vdots\, p_4 & \vdots\, p_5 & \vdots\, p_3 & \vdots\, p_4 & \vdots\, p_5 \\ \vdots\, p_1 & C \vee D & E & E & C \vee D & E & E \\ A \vee B & \multicolumn{2}{c}{E} & \vee E^2 & \multicolumn{2}{c}{E} & \vee E^3 \\ \hline & \multicolumn{5}{c}{E} & \end{array} \vee E^1}{E}$$

Normalisation

By applying all these conversions, mimicking the reduction process of the simple theory of types, we get the following result

Theorem 18.1 (Normalisation)

Each derivation in intuitionistic natural deduction reduces to a normal derivation, in which none of the detour, simplification and permutation conversions can be applied.

Although we are not going the see the details of the proof, since they rely on a complex double induction, we are able to derive a few consequences which are relevant.

Theorem 18.2 (Subformula property)

Let $\pi : \Gamma \vdash A$ be a normal derivation in intuitionistic propositional logic. Then each formula in π is a subformula of some formula in $\Gamma \cup \{A\}$.

Normalisation

By looking at the proof of the Normalisation Theorem,

Corollary 18.3

Let $\pi : \Gamma \vdash A$ be a normal derivation in intuitionistic propositional logic. If A is not atomic or $\bot$, then the last step is an introduction rule.

An immediate consequence is that disjunction is decidable.

Corollary 18.4 (Disjunction property)

Let $\pi : \Gamma \vdash A \vee B$ be a normal derivation in intuitionistic propositional logic. Then, there is a subproof π' of π whose conclusion is either A or B.

Similar results hold for intuitionistic first order logic, and, in particular

Corollary 18.5 (Explicit definability)

Let $\pi : \Gamma \vdash \exists x.A$ be a normal derivation in intuitionistic first order logic. Then, there is a subproof π' of π whose conclusion is either $A[t/x]$ for some term t.

Normalisation

It is important to remark that we have proved these results about normalisation in the natural deduction system for pure propositional logic. Choosing a different deductive system, although sound and complete, does not necessarily lead to the same result.

Also, adding a theory, and, thus, instances of the axiom rule may lead to alternative normalisation procedures, or to systems in which normalisation cannot be obtained.

In these cases, the constructive nature of intuitionistic logic, stemming from Corollaries 18.4 and 18.5, is not automatically achieved.
As an obvious counterexample, consider that classical logic is just intuitionistic logic plus the theory $\{A \vee \neg A \colon A \text{ formula}\}$.

References

The propositions-as-types interpretation and the normalisation theorem are illustrated in many textbooks: the lesson has been adapted from *A.S. Troelstra* and *H. Schwichtenberg*, Basic Proof Theory, Cambridge Tracts in Theoretical Computer Science 43, Cambridge: Cambridge University Press, (1996). An analysis of normalisation can be found in *S. Negri* and *J. Von Plato*, Structural Proof Theory, Cambridge University Press (2001).

A more computer science oriented text is *B.C. Pierce*, Types and Programming Languages, The MIT Press, (2002), ISBN 978-0262-16209-8.

Homotopy Type Theory is introduced in *The Univalent Foundation Program*, Homotopy Type Theory, Institute of Advanced Studies, Princeton, (2013). The chapter on type theory is also a very nice introduction to the theory of dependent types, which extends the simple theory theories in a natural and constructive way.

Mathematical Logic

Lecture 19

Dr Marco Benini

marco.benini@uninsubria.it

Dipartimento di Scienza e Alta Tecnologia
Università degli Studi dell'Insubria

a.a. 2016/17

Syllabus

Limiting results:

- Peano arithmetic
- Induction
- Standard and non-standard models
- Representable entities

Peano arithmetic

Peano arithmetic is the standard formal theory describing natural numbers and their properties.

It is composed by a series of axioms, divided into groups, and it is interpreted in classical first order logic.

The very same theory, interpreted in intuitionistic first order logic is called Heyting arithmetic. Despite they are syntactically identical, their interpretations are quite different. For example, in Peano arithmetic it is possible to show that there are functions which cannot be computed, while every function which can be proved to exist in Heyting arithmetic, is computable, because of the constructive nature of the logic.

Peano arithmetic

Peano arithmetic is based on the language generated by the the signature

$$\langle \{\mathbb{N}\}; \{0 : \mathbb{N}, S : \mathbb{N} \to \mathbb{N}, +, \cdot : \mathbb{N} \times \mathbb{N} \to \mathbb{N}\}; \{= : \mathbb{N} \times \mathbb{N}\} \rangle \ .$$

The first group of axioms defines what is a natural number:

$$\forall x, y. S\, x = S\, y \supset x = y \ ; \tag{1}$$

$$\forall x. S\, x \neq 0 \ . \tag{2}$$

The idea is that natural numbers are the elements of the free algebra generated by 0 and S. The successor function S, given a number x, calculates the next number, $x+1$. So natural numbers are written in the unary representation, and they are naturally equipped with a total order structure with minimum.

Peano arithmetic

The second group of axioms define addition and multiplication:

$$\forall x.0 + x = x \ ; \tag{3}$$

$$\forall x, y.S\,x + y = S(x + y) \ ; \tag{4}$$

$$\forall x.0 \cdot x = 0 \ ; \tag{5}$$

$$\forall x, y.S\,x \cdot y = x \cdot y + y \ . \tag{6}$$

It is worth remarking the inductive nature of these definitions.

Peano arithmetic

The third and last group of axioms is a schema: for any formula A,

$$A[0/x] \wedge (\forall x.\, A \supset A[S\,x/x]) \supset \forall x.\, A \tag{7}$$

This schema formalises induction on the structure of natural numbers:

- if A holds on 0
- and, assuming that A holds on x, we can show that it holds on $S\,x$,
- then, A holds for every $x \in \mathbb{N}$.

Induction

There is a link between induction and recursion: an inductive definition induces a recursive procedure that allows to calculate/generate the defining objects, and vice versa, a recursive procedure induces an inductive definition of its results.

Example 19.1

The axioms (3) and (4) provide a recursive schema that allows to calculate the addition:

$$x + y = \text{if } x = 0 \text{ then } y \text{ else let } x = S\,z \text{ in } S(z + y) \ ;$$

Conversely, we may say that the result of the sum is identified by induction of the first summand.

Standard model

The standard model for Peano arithmetic is the structure which interprets the signature as

- the unique sort into the set of natural numbers, denoted by $\mathbb{N}$;
- the function symbols into the zero number, the successor function, and the usual addition and multiplication, respectively.

Any model, i.e., any pair $(\mathscr{M},\sigma)$ is said to be *standard* when $\mathscr{M}$ is the structure above while no restriction is posed on the evaluation σ of variables. Although it may be confusing, we adopt the standard notation which uses the same symbols to denote the formal elements of the syntax, and their intended interpretation. In any standard model, this convention makes no difference.

Since the purpose of the theory of arithmetic is to characterise the class of standard models, it would be nice if these were the only models of the theory. Unfortunately, this is not the case.

Non-standard models

Definition 19.2 (Non-standard model)

Any structure $\mathcal{N}$ on the language of Peano arithmetic which is not isomorphic to the standard model $\mathcal{M}$ but, for any evaluation σ of variables is a model $(\mathcal{N}, \sigma)$ of Peano arithmetic, is called a *non-standard model*.

In the definition above, an isomorphism between structures $f : \mathcal{N} \to \mathcal{M}$ is

- an invertible function between the universes;
- for each term t, $f([\![t]\!]_{\mathcal{N}}) = [\![t]\!]_{\mathcal{M}}$.

If a non-standard model exists, it means that there is a structure $\mathcal{N}$ which makes Peano arithmetic true but interprets some term into an element e in the universe which cannot be mapped in some natural number.

Notice that the element e must be the image of a term under the interpretation function: so, for example, the real numbers consisting of all the non-negative integers, is **not** a non-standard model, even if it is constructed in a very different way from the naturals (all the reals are a quotient of Cauchy sequences).

Non-standard models

Proposition 19.3

There is a non-standard model for Peano arithmetic.

Proof. (i)

Define $S^0(0) = 0$, and $S^{i+1}(0) = S\,S^i(0)$. Evidently the term $S^n(0)$ gets interpreted in n in any model.

Let $\Sigma_n = \{x \neq S^i(0) \colon i < n\}$ be a collection of formulae, and let $\Sigma = \bigcup_{n\in\mathbb{N}} \Sigma_n$.

Calling $\mathscr{M}$ the structure of the standard model, and defining σ_n such that $\sigma_n(x) = n$, evidently the standard model $(\mathscr{M}, \sigma_n)$ makes Σ_n valid, together with all the axioms of Peano arithmetic.

Thus, any finite $\Xi \subset \Sigma$ has a model, because it is contained in Σ_n for some n. Thus, by the Compactness Theorem 10.1, Σ has a model $(\mathscr{N}, \sigma)$ which makes true also all the axioms of Peano arithmetic. ↪

Non-standard models

↪ Proof. (ii)

In this model, $\sigma(x) \neq n$ for any $n \in \mathbb{N}$ because $[\![S^n(0)]\!]_{\mathcal{N}} = n$ but $x \neq S^n(0)$ occurs in Σ, so, by definition of interpretation, $\sigma(x) \neq [\![S^n(0)]\!]_{\mathcal{N}}$.

Hence, there is an element $k \notin \mathbb{N}$ such that $\sigma(x) = k$. But interpreting x on $\mathscr{M}$ leads to some $n \in \mathbb{N}$, whatever evaluation of variables we may choose. So, any function mapping $\mathscr{N}$ to $\mathscr{M}$ has to be non-invertible on the term x.

Thus, $(\mathscr{M}, \sigma)$ is a model of Peano arithmetic, which is not isomorphic to any standard model, so it is non-standard. □

Discussion

The existence of a non-standard model for Peano arithmetic shows that this theory does not describe **exactly** the natural numbers and their properties which can be expressed in the language. Here, not exactly means not only.

The first thought is to try to *complete* Peano arithmetic to prevent the construction of a model like the $(\mathscr{M}, \sigma)$ above. Clearly, the shape of the proof, using the Compactness Theorem, does not allow to obtain this result in a direct way.

However, it is not evident whether the existence of a non-standard model is disturbing: we cannot use the proof of Proposition 19.3 to write a formula which holds in the non-standard model while is does not in any standard model. In fact, we used this property to synthesise the non-standard model from the standard ones.

Discussion

Of course, we can use a theory to separate the non-standard model from any standard one: this is exactly the purpose of the Σ theory in Proposition 19.3.

But, still, it is not clear whether there is *closed* formula, i.e., a formula with no free variables, allowing to separate standard models from non-standard ones.

This would be crucial, since such a formula ϕ does not depend on the evaluation of variables, thus its truth variable would be defined by the structure of the model only. In a sense, ϕ, if it exists, cannot be provable, even if it is true in any standard model, because it would be false in some non-standard model, thus, by the Soundness Theorem, it cannot be proved.

If such a ϕ exists, it means that we have a way to separate models **within** the theory of Peano arithmetic, just by adding a single axiom, ϕ, or its complement, $\neg\phi$.

Representable entities

Definition 19.4 (Numerals)

Given $n \in \mathbb{N}$, the *numeral* $\overline{n}$ representing n is defined as $\overline{0} \equiv 0$, and $\overline{n+1} \equiv S\ \overline{n}$.

Definition 19.5 (Representation)

A relation $R \subseteq \mathbb{N}^k$ is *representable* in Peano arithmetic if and only if there is a formula ϕ such that

- if $(n_1, \ldots, n_k) \in R$ then $\vdash_{\mathrm{PA}} \phi(\overline{n_1}, \ldots, \overline{n_k})$;
- if $(n_1, \ldots, n_k) \notin R$ then $\vdash_{\mathrm{PA}} \neg\phi(\overline{n_1}, \ldots, \overline{n_k})$;

where $\vdash_{PA}$ means 'provable in Peano arithmetic'.
A function $f : \mathbb{N}^k \to \mathbb{N}$ is *representable* in Peano arithmetic if the relation $R = \{(n_1, \ldots, n_k, m) : m = f(n_1, \ldots, n_k)\}$ is representable.
A set $S \subset \mathbb{N}$ is *representable* in Peano arithmetic if its characteristic function is representable.

Representable entities

Example 19.6

Equality is representable in Peano arithmetic.

Proposition 19.7

If the relation $P, Q \subseteq \mathbb{N}^k$ are representable in Peano arithmetic, so are $\neg P$, $P \wedge Q$, and $P \vee Q$.

Proof.

Since P and Q are representable, there are ϕ_P and ϕ_Q as in Definition 19.5.
So, $(n_1, \ldots, n_k) \in \neg P$ if and only if $(n_1, \ldots, n_k) \notin P$. Thus, $\neg\phi_P$ represents $\neg P$, because $\neg\neg\phi_P(n_1, \ldots, n_k) = \phi_P(n_1, \ldots, n_k)$.
Also, $(n_1, \ldots, n_k) \in P \wedge Q$ if and only if $(n_1, \ldots, n_k) \in P$ and $(n_1, \ldots, n_k) \in Q$.
Thus, $\phi_{P \wedge Q} = \phi_P \wedge \phi_Q$. Similarly, $\phi_{P \vee Q} = \phi_Q \vee \phi_Q$. □

Representable entities

Proposition 19.8

The $\underline{0}$ function is representable.

Proof.

Since $\underline{0}\colon \mathbb{N} \to \mathbb{N}$, we have to find a formula representing $Z = \{(n, m)\colon m = \underline{0}(n)\}$. Consider $\phi_{\underline{0}}(x, y) \equiv (y = 0)$.

- If $(n, m) \in Z$, then $m = \underline{0}(n)$, so $m = 0$. Thus, $\phi_{\underline{0}}(\overline{n}, \overline{m}) \equiv (\overline{m} = \overline{0}) \equiv (\overline{0} = \overline{0})$, so $\vdash_{\mathsf{PA}} \phi_{\underline{0}}(\overline{n}, \overline{m})$, by reflexivity.
- If $(n, m) \notin Z$, then $m \neq \underline{0}(n)$, so $m \neq 0$. Thus, $\overline{m} \equiv S\ \overline{m'}$ and $\phi_{\underline{0}}(\overline{n}, \overline{m}) \equiv (\overline{m} = \overline{0}) \equiv (S\ \overline{m'} = \overline{0})$, so $\vdash_{\mathsf{PA}} \neg\phi_{\underline{0}}(\overline{n}, \overline{m})$, by axiom.

□

Representable entities

Theorem 19.9

All recursive functions are representable in Peano arithmetic.

Corollary 19.10

All recursive sets and relations are representable in Peano arithmetic.

These proofs can be found in *Elliott Mendelson*, Introduction to Mathematical Logic, CRC Press. The proof is by induction on the structure of partial recursive functions and it is far too complex to be detailed here: in fact, it is usually absent in most textbooks.
But it is a constructive proof: given a partial recursive function f, it provides an effective method to build a formula representing f.

References

Peano arithmetic is illustrated in most textbooks about logic.

The relation between induction and recursion is deep and complex: an introduction to this is beyond the scope of the present course. The interested reader could refer to *Benjamin C. Pierce*, Types and Programming Languages, MIT Press, (2002), ISBN 978-0-262-16209-8.

The existence of non-standard models can be shown in many different ways. Proposition 19.3 is adapted from *John Bell* and *Moshé Machover*, A Course in Mathematical Logic, North-Holland, (1977), ISBN 0-7204-28440.

The representation of relations, sets, and functions is taken from *Barry Cooper*, Computability Theory, Chapman & Hall/CRC Mathematics, (2004), ISBN 1-58488-237-9.

Mathematical Logic

Lecture 20

Dr Marco Benini

marco.benini@uninsubria.it

Dipartimento di Scienza e Alta Tecnologia
Università degli Studi dell'Insubria

a.a. 2016/17

Syllabus

Limiting results:

- Gödel's First Incompleteness Theorem
- The idea behind the proof
- Coding terms
- Coding formulae

Induction, again

The induction principle says that, fixed a property $P \subseteq \mathbb{N}$, if $0 \in P$ and, for any $n \in \mathbb{N}$, if $n \in P$ then $n+1 \in P$, then $P = \mathbb{N}$.

Clearly, the induction schema (7) in Peano arithmetic is just an approximation of the real induction principle: since $\left|\wp(\mathbb{N})\right| = 2^{|\mathbb{N}|}$ while the collection of formulae on the language of arithmetic has cardinality $|\mathbb{N}|$, we have not enough formulae to represent all the possible properties.

The gap between what can be formalised and what is the intended meaning about the structure of natural numbers, the induction principle at the first place, is responsible for non-standard models.

Incompleteness theorem

Theorem 20.1 (Gödel's Incompleteness Theorem)

Let T be an effective theory which is consistent, and able to represent all the recursive functions. Then, there is a closed formula G such that $T \nvdash G$ and $T \nvdash \neg G$.

A theory is said to be *effective* when the set of axioms is recursive, that is, applying a *coding* to its axioms so that they become a set of numbers, this set is recursive.
A coding of Peano arithmetic, or, more in general, of recursive functions, is a total map g from the expressions of the syntax (terms, formulae, proofs) to $\mathbb{N}$ such that

- g is injective;
- g is recursive;
- g^{-1} on the image of g is recursive, too.

Strategy

The proof of the incompleteness theorem is complex. It has a difficult part, the fixed point lemma, and a lot of technicalities.

The strategy is to consider the sentence "this sentence is not provable".

- we will show that there is a coding function that maps terms, formulae and proofs into natural numbers;
- hence, it is possible to write a formula which says "there is a number p which is the code of a proof of the sentence x";
- negating that formula, we can express the fact that x is not provable;
- we will show a fixed point theorem saying that there exists a fixed point of the transformation which maps each sentence x to the code of the sentence expressing that x is not provable;
- thus, the sentence G becomes the formula stating that x is not provable with x substituted with the fixed point;
- the meaning of G is that G is not provable;
- but G must be true in the standard model, otherwise the theory would be contradictory, so the result follows.

Coding terms

In the following, for the sake of simplicity, we will assume the set of variables in the language of Peano arithmetic to be $V = \{x_i : i \in \mathbb{N}\}$.

Definition 20.2 (Coding terms)

The *Gödel's coding function* g on terms is inductively defined as follows:

- $g(0) = 2 \cdot 3$;
- $g(x_i) = 2 \cdot 3^2 \cdot 5^{i+1}$;
- $g(S\,t) = 2 \cdot 3^3 \cdot 5^{g(t)}$;
- $g(t+s) = 2 \cdot 3^4 \cdot 5^{g(t)} \cdot 7^{g(s)}$;
- $g(t \cdot s) = 2 \cdot 3^5 \cdot 5^{g(t)} \cdot 7^{g(s)}$.

Thanks to the theorem saying that natural numbers admit a unique factorisation in primes, g is computable, injective, and g^{-1} is computable.

Coding terms

A few remarks are needed:

- each code for a term is of the form $2 \cdot n$, with n odd;
- the exponent of the factor 3 tells whether the term is 0, a variable, a successor, a sum, or a multiplication;
- the parameters of a term, i.e., the index of the variable, or the arguments of the successor, of the sum, or the multiplication, are the exponents of the factors 5 and 7, in that order.

Hence, intuitively, it is possible to write a formula in Peano arithmetic that tells whether its argument is a code of a term. This can be formalised by showing that the set of codes for terms is recursive, so that Proposition 19.10 yields the result.

Coding formulae

Definition 20.3 (Coding formulae)

The *Gödel's coding function* g on formulae extends the coding of terms and it is inductively defined as follows:

- $g(\top) = 2^2 \cdot 3$;
- $g(\bot) = 2^2 \cdot 3^2$;
- $g(t = s) = 2^2 \cdot 3^3 \cdot 5^{g(t)} \cdot 7^{g(s)}$;
- $g(\neg A) = 2^2 \cdot 3^4 \cdot 5^{g(A)}$;
- $g(A \wedge B) = 2^2 \cdot 3^5 \cdot 5^{g(A)} \cdot 7^{g(B)}$;
- $g(A \vee B) = 2^2 \cdot 3^6 \cdot 5^{g(A)} \cdot 7^{g(B)}$;
- $g(A \supset B) = 2^2 \cdot 3^7 \cdot 5^{g(A)} \cdot 7^{g(B)}$;
- $g(\forall x.\, A) = 2^2 \cdot 3^8 \cdot 5^{g(A)} \cdot 7^{g(x)}$;
- $g(\exists x.\, A) = 2^2 \cdot 3^9 \cdot 5^{g(A)} \cdot 7^{g(x)}$.

Again, the coding g is computable, injective, and g^{-1} is computable, too.

Coding formulae

A few remarks are needed:

- each code for a formula is of the form $2^2 \cdot n$, with n odd, so we can separate the codes of terms from the ones of formulae just looking the exponent of the factor 2;
- the exponent of the factor 3 tells which kind of formula the code represents;
- the parameters of a formula are the exponents of the factors 5 and 7, in that order.

Hence, intuitively, it is possible to write a formula in Peano arithmetic that tells whether its argument is a code of a formula. This can be formalised by showing that the set of codes for formulae is recursive, so that Proposition 19.10 yields the result.

Coding sequences

Definition 20.4 (Coding finite sequences)

The *Gödel's coding function* g of a finite sequence $n_1, \ldots, n_k$ of natural numbers is $g(n_1, \ldots, n_k) = 2^3 \cdot \prod_{1 \leq i \leq k} p_{i+1}^{n_i+1}$, with p_j the j-th prime number.

It is clear that the coding function is injective, computable, and its inverse is computable, too. Also, the codes for sequences can be separated by the codes of terms and formulae, and the set of codes for sequences can be represented, in the sense of Proposition 19.10, by some formula of Peano arithmetic, specifically by $\exists y. x = SSSSSSSSS0 \cdot y$.

Coding proofs

Definition 20.5 (Coding proofs)

The *Gödel's coding function* g on proofs extends the previous coding g and it is inductively defined as:

- $g\left(\dfrac{\pi_1 : \Gamma \vdash A \quad \pi_2 : \Gamma \vdash B}{A \wedge B}\,\wedge\mathsf{I}\right) = 2^4 \cdot 3 \cdot 5^{g(\pi_1 : \Gamma \vdash A)} \cdot 7^{g(\pi_2 : \Gamma \vdash B)} \cdot 13^{g(A \wedge B)}$;
- $g\left(\dfrac{\pi : \Gamma \vdash A \wedge B}{A}\,\wedge\mathsf{E}_1\right) = 2^4 \cdot 3^2 \cdot 5^{g(\pi : \Gamma \vdash A \wedge B)} \cdot 13^{g(A)}$;
- $g\left(\dfrac{\pi : \Gamma \vdash A \wedge B}{B}\,\wedge\mathsf{E}_2\right) = 2^4 \cdot 3^3 \cdot 5^{g(\pi : \Gamma \vdash A \wedge B)} \cdot 13^{g(B)}$;
- $g\left(\dfrac{\pi : \Gamma \vdash A}{A \vee B}\,\vee\mathsf{I}_1\right) = 2^4 \cdot 3^4 \cdot 5^{g(\pi : \Gamma \vdash A)} \cdot 13^{g(A \vee B)}$;
- $g\left(\dfrac{\pi : \Gamma \vdash B}{A \vee B}\,\vee\mathsf{I}_2\right) = 2^4 \cdot 3^5 \cdot 5^{g(\pi : \Gamma \vdash B)} \cdot 13^{g(A \vee B)}$;

↪

Coding proofs

↪ (Coding proofs)

- $g\left(\dfrac{\pi_1 : \Gamma \vdash A \vee B \quad \pi_2 : \Gamma, A \vdash C \quad \pi_3 : \Gamma, B \vdash C}{C}\,\vee\mathsf{E}\right) =$
  $2^4 \cdot 3^6 \cdot 5^{g(\pi_1 : \Gamma \vdash A \vee B)} \cdot 7^{g(\pi_2 : \Gamma, A \vdash C)} \cdot 11^{g(\pi_3 : \Gamma, B \vdash C)} \cdot 13^{g(C)}$;
- $g\left(\dfrac{\pi : \Gamma, A \vdash B}{A \supset B}\,\supset\mathsf{I}\right) = 2^4 \cdot 3^7 \cdot 5^{g(\pi : \Gamma, A \vdash B)} \cdot 13^{g(A \supset B)}$;
- $g\left(\dfrac{\pi_1 : \Gamma \vdash A \supset B \quad \pi_2 : \Gamma \vdash A}{B}\,\supset\mathsf{E}\right) = 2^4 \cdot 3^8 \cdot 5^{g(\pi_1 : \Gamma \vdash A \supset B)} \cdot 7^{g(\pi_2 : \Gamma \vdash A)} \cdot 13^{g(B)}$;
- $g\left(\dfrac{\pi : \Gamma, A \vdash \bot}{\neg A}\,\neg\mathsf{I}\right) = 2^4 \cdot 3^9 \cdot 5^{g(\pi : \Gamma, A \vdash \bot)} \cdot 13^{g(\neg A)}$;
- $g\left(\dfrac{\pi_1 : \Gamma \vdash \neg A \quad \pi_2 : \Gamma \vdash A}{\bot}\,\neg\mathsf{E}\right) = 2^4 \cdot 3^{10} \cdot 5^{g(\pi_1 : \Gamma \vdash \neg A)} \cdot 7^{g(\pi_2 : \Gamma \vdash A)} \cdot 13^{g(\bot)}$;
- $g\left(\dfrac{}{\top}\,\top\mathsf{I}\right) = 2^4 \cdot 3^{11} \cdot 13^{g(\top)}$;

↪

Coding proofs

↪ (Coding proofs)

- $g\left(\dfrac{\pi : \Gamma \vdash \bot}{A}\,\bot\mathsf{E}\right) = 2^4 \cdot 3^{12} \cdot 5^{g(\pi : \Gamma \vdash \bot)} \cdot 13^{g(A)}$;
- $g\left(\dfrac{}{A \vee \neg A}\,\mathsf{lem}\right) = 2^4 \cdot 3^{13} \cdot 13^{g(A \vee \neg A)}$;
- $g\left(\dfrac{\pi : \Gamma \vdash A}{\forall x.\, A}\,\forall\mathsf{I}\right) = 2^4 \cdot 3^{14} \cdot 5^{g(\pi : \Gamma \vdash A)} \cdot 13^{g(\forall x.\, A)} \cdot 19^{g(x)}$;
- $g\left(\dfrac{\pi : \Gamma \vdash \forall x.\, A}{A[t/x]}\,\forall\mathsf{E}\right) = 2^4 \cdot 3^{15} \cdot 5^{g(\pi : \Gamma \vdash \forall x.\, A)} \cdot 13^{g(A[t/x])} \cdot 17^{g(t)} \cdot 19^{g(x)}$;
- $g\left(\dfrac{\pi : \Gamma \vdash A[t/x]}{\exists x.\, A}\,\exists\mathsf{I}\right) = 2^4 \cdot 3^{16} \cdot 5^{g(\pi : \Gamma \vdash A[t/x])} \cdot 13^{g(\exists x.\, A)} \cdot 17^{g(t)} \cdot 19^{g(x)}$;
- $g\left(\dfrac{\pi_1 : \Gamma \vdash \exists x.\, A \quad \pi_2 : \Gamma, A \vdash B}{B}\,\exists\mathsf{E}\right) =$
  $2^4 \cdot 3^{17} \cdot 5^{g(\pi_1 : \Gamma \vdash \exists x.\, A)} \cdot 7^{g(\pi_2 : \Gamma, A \vdash B)} \cdot 13^{g(B)} \cdot 19^{g(x)}$;

↪

Coding proofs

↪ (Coding proofs)

- $g\left(\dfrac{}{\forall x.x=x}\,\mathsf{ax}\right) = 2^4 \cdot 3^{18} \cdot 13^{g(\forall x.x=x)} \cdot 19^{g(x)}$;
- $g\left(\dfrac{}{\forall x,y.x=y \supset y=x}\,\mathsf{ax}\right) = 2^4 \cdot 3^{19} \cdot 13^{g(\forall x,y.x=y\supset y=x)} \cdot 19^{g(x,y)}$;
- $g\left(\dfrac{}{\forall x,y,z.x=y \wedge y=z \supset x=z}\,\mathsf{ax}\right) =$
  $2^4 \cdot 3^{20} \cdot 13^{g(\forall x,y,z.x=y\wedge y=z\supset x=z)} \cdot 19^{g(x,y,z)}$;
- $g\left(\dfrac{\pi_1 : \Gamma \vdash A[t/x] \quad \pi_2 : \Gamma \vdash t=r}{A[r/x]}\,\mathsf{ax}\right) =$
  $2^4 \cdot 3^{21} \cdot 5^{g(\pi_1 : \Gamma\vdash A[t/x])} \cdot 7^{g(\pi_2 : \Gamma\vdash t=r)} \cdot 13^{g(A[r/x])} \cdot 19^{g(x)}$;
- $g\left(\dfrac{}{\forall x_1,\ldots,x_n.\exists! z.z=f(x_1,\ldots,x_n)}\,\mathsf{ax}\right) =$
  $2^4 \cdot 3^{22} \cdot 13^{g(\forall x_1,\ldots,x_n.\exists! z.z=f(x_1,\ldots,x_n))} \cdot 17^{g(f(x_1,\ldots,x_n))} \cdot 19^{g(x_1,\ldots,x_n,z)}$;
- $g\left(\dfrac{}{\forall x.Sx\neq x}\,\mathsf{ax}\right) = 2^4 \cdot 3^{23} \cdot 13^{g(\forall x.Sx\neq x)} \cdot 19^{g(x)}$;

↪

Coding proofs

↪ (Coding proofs)

- $g\left(\dfrac{}{\forall x,y.\,Sx=Sy\supset x=y}\,\mathsf{ax}\right)=2^4\cdot 3^{24}\cdot 13^{g(\forall x,y.\,Sx=Sy\supset x=y)}\cdot 19^{g(x,y)}$;
- $g\left(\dfrac{}{\forall x.\,0+x=x}\,\mathsf{ax}\right)=2^4\cdot 3^{25}\cdot 13^{g(\forall x.\,0+x=x)}\cdot 19^{g(x)}$;
- $g\left(\dfrac{}{\forall x,y.\,Sx+y=S(x+y)}\,\mathsf{ax}\right)=2^4\cdot 3^{26}\cdot 13^{g(\forall x,y.\,Sx+y=S(x+y))}\cdot 19^{g(x,y)}$;
- $g\left(\dfrac{}{\forall x.\,0\cdot x=0}\,\mathsf{ax}\right)=2^4\cdot 3^{27}\cdot 13^{g(\forall x.\,0\cdot x=0)}\cdot 19^{g(x)}$;
- $g\left(\dfrac{}{\forall x,y.\,Sx\cdot y=x\cdot y+y)}\,\mathsf{ax}\right)=2^4\cdot 3^{28}\cdot 13^{g(\forall x,y.\,Sx\cdot y=x\cdot y+y))}\cdot 19^{g(x,y)}$;
- $g\left(\dfrac{}{A[0/x]\wedge(\forall x.\,A\supset A[Sx/x])\supset\forall x.\,A}\,\mathsf{ax}\right)=$
  $2^4\cdot 3^{29}\cdot 5^{g(A)}\cdot 13^{g(A[0/x]\wedge(\forall x.\,A\supset A[Sx/x])\supset\forall x.\,A)}\cdot 19^{g(x)}$;
- if $A\in\Gamma$ is a proof by assumption, $g(A)=2^4\cdot 3^{30}\cdot 5^{g(A)}\cdot 7^{g(\Gamma)}\cdot 13^{g(A)}$ with $\Gamma=\{\gamma_1,\ldots,\gamma_n\}$ and $g(\Gamma)=g(\gamma_1,\ldots,\gamma_n)$.

↪

Coding proofs

↪ (Coding proofs)

It should be remarked that $g(e_1,\ldots,e_n)$, when e_i are not numbers should be read as $g(g(e_1),\ldots,g(e_n))$, i.e., the code of the sequence of codes of the elements.

Although it is long and tedious to verify, g is injective, computable, and g^{-1} is recursive. Also, the coding function is written down to make easy to tell pieces apart. For example, the code of the conclusion is always the exponent of the 13 factor.

Numeral

Definition 20.6 (Numeral)

The *numeral* $\ulcorner A \urcorner$ of a formula A is defined as $\ulcorner A \urcorner = S^{g(A)}(0)$, that is, the code of A written in the syntax of Peano arithmetic.

Similarly, the numeral of a term t is $\ulcorner t \urcorner = S^{g(t)}(0)$, the numeral of a proof π is $\ulcorner \pi \urcorner = S^{g(\pi)}(0)$, and the numeral of a sequence is $\ulcorner e_1, \ldots, e_n \urcorner = S^{g(e_1,\ldots,e_n)}(0)$.

Numerals allow to *internalise* the codes: we can, indirectly, speak of a formula (term, proof, sequence) by stating a property of its code. As soon as the property does not rely on the value, but on the "meaning" of the code, this is a perfectly reasonable way to proceed.

Fixed point lemma

Lemma 20.7 (Fixed point)

Let Ξ be a theory in which every (primitive) recursive function is representable, and let A be a formula such that $\mathrm{FV}(A) = \{y\}$. Then, there is a formula δ_A such that $\mathrm{FV}(\delta_A) = \emptyset$ and $\vdash_\Xi \delta_A = A[\ulcorner \delta_A \urcorner / y]$.

Proof. (i)

Let $\Delta_{\mathscr{F}}$ be the map from formulae to formulae defined by $\Delta_{\mathscr{F}}(B) \equiv \exists x_1 . x_1 = \ulcorner B \urcorner \wedge B$. This function is total, computable and injective.

Thus, the map $\Delta_{\mathbb{N}}$ defined by $\Delta_{\mathbb{N}}(g(B)) = g(\Delta_{\mathscr{F}}(B))$ is total on the image of g, (primitive) recursive, and injective.

By hypothesis, there is a formula Δ with $\mathrm{FV}(\Delta) = \{x, y\}$ such that Δ represents the function $\Delta_{\mathbb{N}}$.

Let $F \equiv \exists y . \Delta[x_1/x] \wedge A$. Clearly, $\mathrm{FV}(F) = \{x_1\}$. Also, let $\delta_A = \Delta_{\mathscr{F}}(F)$, that is, $\delta_A \equiv \exists x_1 . x_1 = \ulcorner F \urcorner \wedge F$. Thus, $\mathrm{FV}(\delta_A) = \emptyset$. ↪

Fixed point lemma

↪ Proof. (ii)

Since $\exists y.\Delta[\ulcorner F \urcorner/x] \wedge A$ implies $\exists x_1, y.\Delta[x_1/x] \wedge A$ with $x_1 = \ulcorner F \urcorner$, we can prove that $\exists x_1.x_1 = \ulcorner F \urcorner \wedge \exists y.\Delta[x_1/x] \wedge A$, which is just δ_A, Hence, we have shown that $\vdash (\exists y.\Delta[\ulcorner F \urcorner/x] \wedge A) \supset \delta_A$.

Conversely, $\delta_A \equiv \exists x_1.x_1 = \ulcorner F \urcorner \wedge \exists y.\Delta[x_1/x] \wedge A$, so δ_A implies $\exists x_1, y.\Delta[x_1/x] \wedge A$ with $x_1 = \ulcorner F \urcorner$, thus we can prove that $\exists y.\Delta[\ulcorner F \urcorner/x] \wedge A$. Hence, we have shown that $\vdash \delta_A \supset (\exists y.\Delta[\ulcorner F \urcorner/x] \wedge A)$, thus δ_A and $\exists y.\Delta[\ulcorner F \urcorner/x] \wedge A$ are equivalent.

But Δ represents $\Delta_{\mathbb{N}}$, so Ξ allows to prove, for each $n \in \mathbb{N}$, $\vdash A[S^n(0)/x] = (y = S^{\Delta_{\mathbb{N}}(n)}(0))$. Specialising to $n = g(F)$, we obtain $\vdash \forall y.\Delta[\ulcorner F \urcorner/x] = (y = S^{\Delta_{\mathbb{N}}(g(F))}(0))$. ↪

Fixed point lemma

↪ Proof. (iii)

So the previous equivalence $\vdash \delta_A = (\exists y.\, \Delta[\ulcorner F \urcorner / x] \wedge A)$ allows to derive $\vdash \delta_A = (\exists y.\, y = S^{\Delta_{\mathbb{N}}(g(F))}(0) \wedge A)$.

Evidently, we can prove $\vdash A\left[S^{\Delta_{\mathbb{N}}(g(F))}(0)/y\right] = \left(\exists y.\, y = S^{\Delta_{\mathbb{N}}(g(F))}(0) \wedge A\right)$, thus we can immediately prove $\vdash \delta_A = A\left[S^{\Delta_{\mathbb{N}}(g(F))}(0)/y\right]$.

But $\ulcorner \delta_A \urcorner = S^{g(\delta_A)}(0) = S^{g(\Delta_{\mathscr{F}}(F))}(0) = S^{\Delta_{\mathbb{N}}(g(F)))}(0)$. Thus, the proof above can be rephrased as $\vdash \delta_A = A[\ulcorner \delta_A \urcorner / y]$. □

Provability predicate

Definition 20.8 (Provability predicate)

The formula $\mathscr{D}$ with $\mathrm{FV}(\mathscr{D}) = \{x, y\}$ is defined as
$\mathscr{D} \equiv \exists z.13^y \cdot z = x \wedge \mathrm{isExpr}(x) \wedge \mathrm{isExpr}(y) \wedge \mathrm{isProof}(x) \wedge \mathrm{isFormula}(y)$.

The *provability predicate* T is the formula $\exists x.\mathscr{D}$, having $FV(T) = \{y\}$.

Clearly, $\mathscr{D}[\ulcorner\pi\urcorner/x, \ulcorner A\urcorner/y]$ holds exactly when A is the conclusion of the proof $\pi\colon \vdash A$. And, consequently, $T[\ulcorner A\urcorner/y]$ holds when A is provable.

The formulae isExpr(x), isExpr(y), isProof(x), and isFormula(y) in the definition of $\mathscr{D}$ have not been made explicit. While isProof(x) can be defined as $\exists z.2^4 \cdot z = x$, and isFormula(y) can be defined as $(\exists z.2^3 \cdot z = x) \wedge \neg$isProof(x), the definition of isExpr comes from the fact that the collection of codes forms a recursive set. It could be written down in an explicit way, but it is a cumbersome formula.

Incompleteness theorem

Theorem 20.9 (Gödel's Incompleteness Theorem)

Let T be an effective theory which is consistent, and able to represent all the recursive functions. Then, there is a closed formula G such that $T \nvdash G$ and $T \nvdash \neg G$.

Proof.

Consider the formula $\neg T[x/y]$: applying the fixed point lemma, there is G such that $\mathrm{FV}(G) = \emptyset$ and $\vdash G = \neg T[\ulcorner G \urcorner / y]$.

Assume there is $\pi : \vdash G$. Then $\vdash \neg T[\ulcorner G \urcorner / y]$. But, because $\pi : \vdash G$, it holds that $\vdash \mathscr{D}[\ulcorner \pi \urcorner / x, \ulcorner G \urcorner / y]$, and thus $\vdash \exists x.\mathscr{D}[\ulcorner G \urcorner / y]$, that is, $\vdash T[\ulcorner G \urcorner / y]$, making the theory non consistent. Hence $\nvdash G$.

Oppositely, suppose there is $\pi : \vdash \neg G$. Then $\vdash T[\ulcorner G \urcorner / y]$ by definition of G, so $\vdash \exists x.\mathscr{D}[\ulcorner G \urcorner / y]$. But this means that there exists $\theta : \vdash G$ with $x = \ulcorner G \urcorner$. Thus, again, we get a contradiction. Thus $\nvdash \neg G$. □

References

The original proof of the first incompleteness theorem can be found in *Kurt Gödel*, Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme, I, Monatshefte für Mathematik und Physik 38, 173–198, (1931).

The proof has been generalised and polished by Rosser, and we have shown a slightly reworked version of Rosser's result. The reference is *John Barkley Rosser*, Extensions of some theorems of Gödel and Church, Journal of Symbolic Logic 1, 87–91 (1936).

An account can be found in *John Bell* and *Moshé Machover*, A Course in Mathematical Logic, North-Holland, (1977), ISBN 0-7204-28440. Nevertheless, the lecture has been prepared roughly following some unpublished notes from the course held by Silvio Valentini in 1991.

Mathematical Logic

Lecture 21

Dr Marco Benini

marco.benini@uninsubria.it

Dipartimento di Scienza e Alta Tecnologia
Università degli Studi dell'Insubria

a.a. 2016/17

Syllabus

Limiting results:

- Gödel's Second Incompleteness Theorem
- Meaning and consequences

Properties of provability

Proposition 21.1

For any pair of formulae A and B in Peano arithmetic,

1. $\vdash T[\ulcorner A \urcorner / y]$ *if and only if* $\vdash A$;
2. $\vdash T[\ulcorner A \supset B \urcorner / y] \wedge T[\ulcorner A \urcorner / y] \supset T[\ulcorner B \urcorner / y]$;
3. $\vdash T[\ulcorner A \urcorner / y] = T[\ulcorner T[\ulcorner A \urcorner / y] \urcorner / y]$;
4. $\vdash (T[\ulcorner A \urcorner / y] \wedge T[\ulcorner B \urcorner / y]) = T[\ulcorner A \wedge B \urcorner / y]$;
5. *if* $\vdash A \supset B$ *then* $\vdash T[\ulcorner A \urcorner / y] \supset T[\ulcorner B \urcorner / y]$;
6. *if* $\vdash (T[\ulcorner A \urcorner / y] \wedge A) \supset B$, *then* $\vdash T[\ulcorner A \urcorner / y] \supset T[\ulcorner B \urcorner / y]$.

These properties, we are not going to prove, show that the provability predicate T allows (i) to prove A whenever there is proof the A is provable; (ii) it acts naturally with respect to implication and conjunction; (iii) proving provability is equivalent to prove that provability is provable.

Properties of provability

Proposition 21.2

In Peano arithmetic, if $\vdash A = \neg T[\ulcorner A \urcorner / y]$*, then* $\vdash T[\ulcorner A \urcorner / y] = T[\ulcorner \bot \urcorner / y]$.

Again, without proving it, the proposition says that every formula, which behaves like Gödel's G, is provable if and only if $\bot$ is provable, a fact that captures the content of Theorem 20.9. But, and this is important, the proposition proves that this fact holds **inside** the theory, which is not obvious.

Second incompleteness theorem

Theorem 21.3 (Gödel's second incompleteness theorem)

There is no provable formula C in Peano arithmetic which codes the consistency of the theory, i.e., such that $\vdash C \supset \neg T[\ulcorner \bot \urcorner / y]$.

Proof.

Suppose there is C such that $\vdash C$ and $\vdash C \supset \neg T[\ulcorner \bot \urcorner / y]$. Then, $\vdash \neg T[\ulcorner \bot \urcorner / y]$, which means that $\bot$ is not provable, that is, Peano arithmetic cannot contain a contradiction, hence it is consistent.

From Theorem 20.9, there is a formula G such that $\vdash G = \neg T[\ulcorner G \urcorner / y]$, but $\nvdash G$. By Proposition 21.2, $\vdash T[\ulcorner G \urcorner / y] = T[\ulcorner \bot \urcorner / y]$, so $\nvdash \neg T[\ulcorner \bot \urcorner / y]$. Thus, we have a contradiction, showing that C cannot exist. □

Mathematical meaning

The incompleteness theorems closes the quest for a universal, self-contained foundation of Mathematics which is able to prove its own consistency. Simply, such a system cannot exist.

Nevertheless, these theorems opened the way to many developments, and to some of the other fundamental results in XX^{th} century:

- the effective construction of non-computable functions
- the idea of coding lead to reason "modulo a coding function", which has been influential in algebra, algebraic geometry, algebraic topology, number theory, ...
- examples of independent statements arose in many fields, and they shed lights to a variety of hidden aspects of apparently clean notions, like, for example, the assumptions behind cardinality in set theory

Foundational consequences

Having a mathematical theory T which is powerful enough to represent Peano arithmetic has the consequence that we cannot prove its consistency within T. We need a theory T', containing T, and more powerful.

This fact led to the development of many hierarchies of formal systems to classify the power of mathematical theories: we scratched just the surface, by showing that the consistency of Peano arithmetic can be proved in a stronger system. But, how much stronger? Since the proof of Gödel's results, much deeper analyses have been conducted, and nowadays this part of Logic is a complex, intricate, difficult field on its own.

In constructive mathematics, the same fact led to doubt that "truth" is the right concept to analyse, and there are approaches favouring the notion of provability as the real foundation of Mathematics. This has a number of consequences, which we do not want to discuss here.

Understanding

For a very long time, mathematicians regarded the incompleteness theorems as strange beasts: something which is important, but, essentially, with no influence in the mathematical practise.

For example, the textbook of Bell and Machover we referred to many times, explicitly says that the sentences which are not provable in Peano arithmetic are not important in arithmetic, because they have no "arithmetical" content, but just a logical content. This is true for the sentence G, and for most other sentences we can construct within the logical analysis.

Unfortunately, there are purely arithmetical properties of genuine interest for mathematicians not working in logic, which are independent from Peano arithmetic.

Natural incompleteness

Theorem 21.4 (Paris, Harrington)

For all $e, r, k \in \mathbb{N}$, there is $M \in \mathbb{N}$ such that, for every $f : \{F \subseteq \{0, \ldots, M\} : |F| = e\} \to \{0, \ldots, r\}$, there is $H \subseteq \{0, \ldots, M\}$ such that

- $|H| \geq \max\{k, \min H\}$, *and*
- *exists $v \leq r$ such that, for all $F \subseteq H$ with $|F| = n$, $f(x) = v$ for each $x \in F$.*

By using the Infinite Ramsey Theorem, it is not too difficult to derive a value $M \in \mathbb{N}$ which makes the statement true on naturals. This proof is carried out either in second-order arithmetic, with the full induction principle, or in a suitable set theory, e.g., **ZFC**. Nevertheless, it is possible to show, *within Peano arithmetic*, that the combinatorial principle in Theorem 21.4 implies the consistency of Peano arithmetic, thus it is impossible to prove in that theory, according to Gödel's second incompleteness theorem.

Natural incompleteness

Actually, a simplified version of Theorem 21.4 suffices:

Theorem 21.5
For all $n \in \mathbb{N}$, there is $M \in \mathbb{N}$ such that, for every function $f : \{F \subseteq \{0,\dots,M\} : |F| = n\} \to \{0,1\}$, there is $H \subseteq \{0,\dots,M\}$ for which, for all $F \subseteq H$ with $|F| = n$, $f(F) = \{0\}$, and $|H| > n(2^{n \min H} + 1)$.

This theorem and the previous one are *natural* in the sense that, changing the first condition in Theorem 21.4 to $|H| \geq k$, we get the Finite Ramsey Theorem, which is provable inside Peano arithmetic, and which is the starting point for a large branch of Combinatorics.

Natural incompleteness

Another important theorem from a different branch of combinatorics is independent from Peano arithmetic: it holds in the standard model, but we cannot prove it in the theory. This is the famous Kruskal's theorem on trees. A simplified version suffices to yield the independence result.

Theorem 21.6

There is some $n \in \mathbb{N}$ such that, if $T_1, \ldots, T_n$ is a finite sequence of trees, where T_k has $k+n$ vertices, then, for some $i < j$, there is an injective map $f : T_i \to T_j$ between the vertices of the trees which preserves paths.

The independence proof for this theorem follows a different pattern: it is possible to show that any function which provably exists in Peano arithmetic cannot grow too fast, but the above theorem allows to construct a function which grows even faster. And this suffices to establish the fact that the theorem is unprovable in Peano arithmetic.

Natural incompleteness

Kruskal's Theorem plays an important role in the algebra of well quasi orders, a topic which has shown relevance in proving the termination of algorithms, so the above independence result has a direct, negative, application to Computer Science, for example.

In this sense, Kruskal's Theorem is "natural" and practically significant.

Incompleteness in set theory

We have already discussed how the Axiom of Choice, the Continuum Hypothesis, and the Generalised Continuum Hypothesis are independent from **ZF**. All these statements are "natural", as they state properties of sets which are inherently of interest, either because of their consequences, or because they impose a regular structure over the objects we want to study.

In fact, the independence results in set theory and in Peano arithmetic are related. For example, Theorem 21.4 is a restriction to the finite case of the proof of independence about the existence of large cardinals.

A cardinal k is said to be *large*, simplifying a bit, when, for every $x \in k$, $\wp x \in k$, too. This fact is spelt as, k is large when, for every $f : \{\{k_1, k_1\} : k_1 \in k, k_2 \in k\} \to 2$, there is $\lambda \in k$ such that f restricted to λ is constant.

Ordinal analysis

There is a branch of proof theory devoted to study the "power" of deductive systems, showing which is the minimal ordinal to which transfinite induction can be relativised in order to prove a consistency statement.

This is a deep, delicate, difficult, and complex part of logic, still in development: it is sometimes referred to as "reverse mathematics" when the goal is to find the minimal theory in which a given statement can be shown to hold.

References

The technical proof of the second incompleteness theorem can be found, for example, in *John Bell* and *Moshé Machover*, A Course in Mathematical Logic, North-Holland, (1977), ISBN 0-7204-28440.

The discussion is general, and there is no specific reference for it. Some ideas could be found in *Jon Barwise*, Handbook of Mathematical Logic, Studies in Logic and the Foundations of Mathematics 90, North-Holland, (1977), ISBN 0-444-863888-5.

As an example of a (very) popular book which deals with incompleteness, we signal *D. Hofstadter*, Gödel, Escher, Bach: an Eternal Golden Braid, Basic books, (1979), ISBN 0-465-02656-7. It is an enjoyable account for non-specialists, but it also contains many debatable points and opinions. Nevertheless, the mathematical content is, essentially, precise—and the author won the Pulitzer prize for non-fiction.

References

A dated, but still valid reference for Ramsey theory is *R. Graham, B. Rothschild, J.H. Spencer*, Ramsey Theory, 2nd edition, John Wiley and Sons, (1990), ISBN 0-4715-0046-1.

The original paper *J.B. Kruskal*, Well-quasi-ordering, the tree theorem, and Vazsonyi's conjecture, Transactions of the American Mathematical Society 95(2), pp. 210–225, American Mathematical Society, (1960), is an inspiring introduction to the theorem and its motivation.

Although there are many texts providing a general overview of combinatorics, my preferred one is *M. Bóna*, A Walk Through Combinatorics, 2nd edition, World Scientific, (2006), ISBN 981-256-885-9.

References

The link between Kruskal's theorem and logic is analysed in depth in *J.H. Gallier*, What's so special about Kruskal's theorem and the ordinal Γ_0? A survey of some results in proof theory, Annals of Pure and Applied Logic 53(3), pp. 199-260, (1991).

The original publication about the Paris-Harrington theorem can be found in *Jon Barwise*, Handbook of Mathematical Logic, Studies in Logic and the Foundations of Mathematics 90, North-Holland, (1977), ISBN 0-444-863888-5.

Finally, a fine introduction to ordinal analysis can be found in *Michael Rathjen*, The art of ordinal analysis, Proceedings of the International Congress of Mathematicians, volume 2, pp. 45–70, (2006), ISBN 978-3-03719-022-7, written by a master of the field.

The end