B1.1 Logic

Martin Bays

Oxford, MT 2024

1. Introduction

1.1. Historical motivation

- In the 19th century, the need for conceptual foundations in analysis became clear, leading to attempts to formalise notions of **infinity, infinitesimal, limit, ...**

  "*The definitive clarification of the nature of the infinite has become necessary, not merely for the special interests of the individual sciences but for the honour of human understanding itself.*"
  – Hilbert 1926

- Hilbert's 2nd Problem, 1900 ICM address: Prove consistency of an axiom system for arithmetic.

  > "*I am convinced that it must be possible to find a direct proof for the compatibility of the arithmetical axioms.*"
  > – Hilbert 1900

- Early attempts to formalise mathematics:
  - *Cantor*'s **naive** set theory;
  - *Frege*'s Begriffsschrift and Grundgesetze.

  For any expressible property $P(x)$, Frege's system posited the existence of the set

$$\{x : P(x)\}.$$

- **Russell's paradox**:
  consider the set $R := \{s : s \notin s\}$

$$\begin{array}{lcll} R \in R & \Rightarrow & R \notin R & \text{contradiction} \\ R \notin R & \Rightarrow & R \in R & \text{contradiction} \end{array}$$

  $\rightsquigarrow$ *Fundamental crisis in the foundations of mathematics.*

1.2. Hilbert's Programme

Step 1. Find a uniform formal **language** for all mathematics.

Step 2. Find a complete **proof system** for deducing consequences of axioms.

Step 3. Find a complete system of **axioms** for mathematics.

Step 4. Prove **consistency** of the resulting system, i.e. that it does not lead to contradictions.

Where:

- **Complete** in 2+3 would mean: every mathematical statement admits a proof or disproof in the system.
  ("*Wir müssen wissen. Wir werden wissen.*"*)

- The system should be **finitary**, i.e. effective/computable/algorithmic,
  so e.g. you can't just take as axioms *all true mathematical statements*.

**We must know. We will know.*

1.3. Results on Hilbert's programme

Step 1. (Formal language for mathematics):
Possible in the framework of
ZF = *Zermelo-Fraenkel set theory* or
ZFC = **ZF** + *Axiom of* **C***hoice*.
(Covered in B1.2)

Step 2. (Complete proof system):
Possible in **1st-order logic**, by
Gödel's Completeness Theorem.
(Covered in B1.1 - this course)

Step 3. (Complete axiom system):
Not possible, by
Gödel's 1st Incompleteness Theorem:
there is no computable axiomatisation
of arithmetic.
(Covered in C1.2)

Step 4. (Proving consistency):
Not possible, by
Gödel's 2nd Incompleteness Theorem:
a sufficiently powerful consistent system
can not prove its own consistency.
(Covered in C1.2)

1.4. Successes of mathematical logic

In summary, the positive outcomes from Steps 1 and 2 left us with a form of mathematical logic with which we can:

- Provide a uniform, unambiguous **language** for mathematics.
- Give a precise formal definition of a **proof**.
- Explain and guarantee **exactness, rigour, and certainty** in mathematics.
- Establish the **foundations** of mathematics.

B1 (Foundations)
= B1.1 (Logic) + B1.2 (Set theory)

1.5. Decidability

Step 3. of Hilbert's programme fails:
there is no computable axiomatisation
for the entire body of mathematics
(or even just of arithmetic).

But: many important *parts* of mathematics *are* completely and computably axiomatizable; they are **decidable**, i.e. there is an *algorithm = program = effective procedure* to decide whether a sentence is true or false.

Example: $\mathrm{Th}(\mathbb{C};+,\cdot)$, the **1st-order theory** of the field $\mathbb{C}$.

Axioms = *field axioms*
+ *all non-constant polynomials have a zero*
+ *the characteristic is* 0.

Every **algebraic** property of $\mathbb{C}$ follows from these axioms.
Similar results hold for e.g. the real field and for vector spaces.
$\rightsquigarrow$ C1.1 Model Theory.

1.6. Why *mathematical* logic?

- The language and deduction rules are tailored for *mathematical objects* and mathematical ways of reasoning.

- The *method* is mathematical. Formulas expressing mathematical statements, as well as proofs of such statements, will themselves be defined as finitary mathematical objects. We will reason about them with ordinary mathematical techniques, of the same kind we use to reason about natural numbers.

- Logic has *applications* in other areas of mathematics, as well as in theoretical computer science.

1.7. Outline of the course

The main result of this course is *Gödel's Completeness Theorem* for first-order logic, which shows that every consequence of given mathematical axioms admits a proof from those axioms. We first study the simpler case of propositional logic, and prove the corresponding completeness theorem there. We end the course by applying our results to some familiar mathematical structures.

Part I: Propositional Logic

We begin by studying **propositional logic**, which deals with statements built out of simpler ones using connectives such as "and", "or", and "not". This is not in itself adequate for formalising mathematics, but we will later refine it to first-order logic, which is. We first consider propositional logic in isolation, then in Part II we extend our treatment to full first-order logic.

Example 1.1. Propositional logic formalises deductions of the following kind:

- 1. Socrates is alive or Socrates is dead.
  2. Socrates is not alive.
  Therefore: Socrates is dead.

- 1. If Socrates is a vampire and vampires are immortal, then Socrates is not dead.
  2. Socrates is dead.
  Therefore: Either Socrates is not a vampire, or vampires are not immortal.

To preview the formalism, we will write these respectively as:

- $\{(p_0 \vee p_1), \neg p_0\} \models p_1$.

- $\{((p_2 \wedge p_3) \to \neg p_1), p_1\} \models (\neg p_2 \vee \neg p_3)$.

We use variables to denote propositions - e.g. p_0 for "Socrates is alive".
A *proposition* is something which can be true or false.

2. Syntax

We define a **language** $\mathcal{L}_{\mathsf{prop}}$ for propositional logic.

2.1. Strings

Definition 2.1. The **alphabet** of $\mathcal{L}_{\mathsf{prop}}$ consists of the following abstract symbols:

$$\neg, \rightarrow, \wedge, \vee, \leftrightarrow, (,), p_0, p_1, p_2, \ldots .$$

The p_i are called **propositional variables**, and $\mathcal{L}_{\mathsf{prop}}$ has one propositional variable p_i for each natural number i.

Definition 2.2. A **string** of $\mathcal{L}_{\text{prop}}$ is any finite sequence of symbols from the alphabet of $\mathcal{L}_{\text{prop}}$.

Example 2.3.

$$\begin{array}{rl} \text{(i)} & \to p_{13}() \\ \text{(ii)} & ((p_0 \wedge p_1) \to \neg p_2) \\ \text{(iii)} &))\neg)p_{37} \end{array}$$

Definition 2.4. The **length** len(A) of a string A is the number of symbols in it.

So the strings in the examples have lengths 4, 10, and 5 respectively.
(A propositional variable is considered as a single symbol.)

2.2. Formulas

Definition 2.5. A **formula** of $\mathcal{L}_{\text{prop}}$ is a string of one of the following forms:

I. p_i, where $i \in \mathbb{N}$.

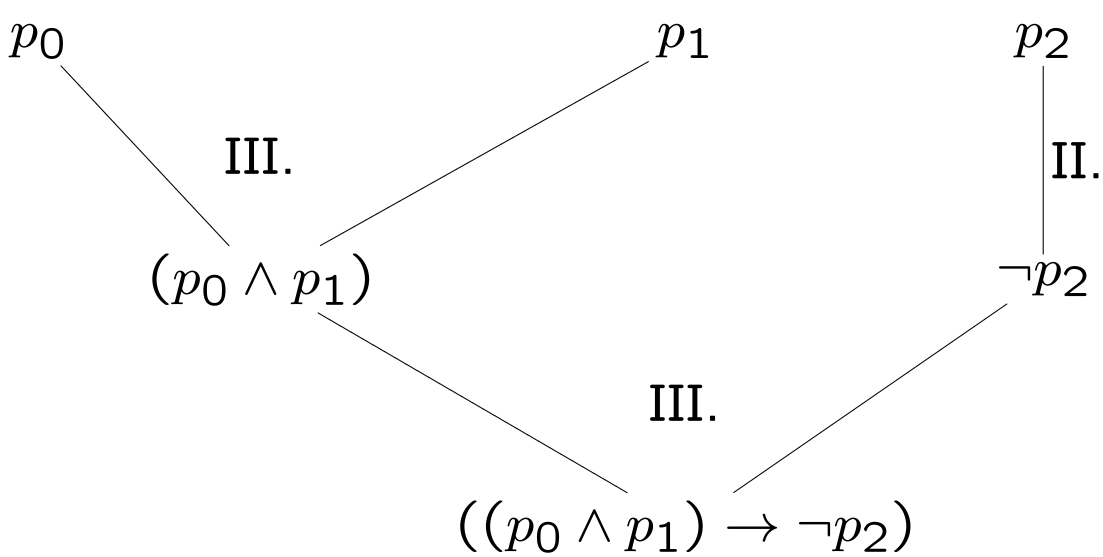
II. $\neg\phi$, where ϕ is a formula.

III. $(\phi \to \psi)$ or $(\phi \wedge \psi)$ or $(\phi \vee \psi)$ or $(\phi \leftrightarrow \psi)$, where ϕ and ψ are both formulas.

In other words, a string ϕ is a formula if and only if ϕ can be obtained from propositional variables by finitely many applications of the *formation rules* II. and III.

Example 2.6. The string $((p_0 \wedge p_1) \to \neg p_2)$ is a formula.

Proof:



$\square$

Parentheses are important, e.g.
$(p_0 \wedge (p_1 \to \neg p_2))$ is a different formula and
$p_0 \wedge (p_1 \to \neg p_2)$ is not a formula at all.

The formulas are the strings which make "grammatical sense", and we will soon define a *semantics* under which they "mean something".

Corresponding to these formation rules, we call $\neg$ a "unary connective", and $\rightarrow, \wedge, \vee, \leftrightarrow$ "binary connectives". We summarise their pronunciation and terminology in the following table.

$\neg\phi$	"not ϕ"	*negation*
$(\phi \rightarrow \psi)$	"ϕ implies ψ"	*implication*
$(\phi \wedge \psi)$	"ϕ and ψ"	*conjunction*
$(\phi \vee \psi)$	"ϕ or ψ"	*disjunction*
$(\phi \leftrightarrow \psi)$	"ϕ if and only if ψ"	*equivalence*

Example 2.7. The strings $\to p_{17}()$ and $))\neg)p_{32}$ are not formulas.

Indeed, if ϕ is a formula, then ϕ is of one of the forms I., II, or III., and in particular one of the following must hold:

1. ϕ is a propositional variable.

2. The first symbol of ϕ is $\neg$.

3. The first symbol of ϕ is (.

Theorem 2.8 (The unique readability theorem)**.**
A formula can be constructed in only one way:
For each formula ϕ **exactly one** *of the following holds*

(a) ϕ *is* p_i *for some unique* $i \in \mathbb{N}$*;*

(b) ϕ *is* $\neg\psi$ *for some* **unique** *formula* ψ*;*

(c) ϕ *is* $(\psi \star \chi)$ *for some* **unique** *pair of formulas* ψ*,* χ *and a* **unique** *binary connective* $\star \in \{\rightarrow, \wedge, \vee, \leftrightarrow\}$*.*

Proof: Problem sheet 1.

2.3. Countability

Recall that a set X is **countable** if $X = \emptyset$ or there are x_i for $i \in \mathbb{N}$ such that $X = \{x_0, x_1, \ldots\}$, i.e. there exists a surjection $\mathbb{N} \to X$.

Let Form($\mathcal{L}_{\text{prop}}$) be the set of all formulas of $\mathcal{L}_{\text{prop}}$.

Fact 2.9. Form($\mathcal{L}_{\text{prop}}$) *is countable.*

This will be proven formally in B1.2 Set Theory, as a consequence of the basic axioms ZF of set theory. More precisely, it will be proven that the set of finite strings in a countable alphabet is countable, and that any subset of a countable set is countable.

3. Semantics

3.1. Valuations

In natural language, the **truth** or **falsity** of a sentence using logical connectives is determined by the truth or falsity of its subclauses:
"Socrates is dead or Socrates is a vampire" is true if "Socrates is dead" is true.

Propositional logic abstracts this to a recursive definition of the **truth value** T ('true') or F ('false') of a formula ϕ in terms of the truth values of the propositional variables occurring in ϕ.

Definition 3.1. A **valuation** v is a function

$$v : \{p_0, p_1, p_2, \ldots\} \to \{T, F\}.$$

Given a valuation v we extend v uniquely to a function

$$\widetilde{v} : \text{Form}(\mathcal{L}_{\text{prop}}) \to \{T, F\}.$$

defined recursively as follows.

Suppose ϕ is a formula, and $\widetilde{v}$ has been defined on formulas of length $< \text{len}(\phi)$.

We split into the cases given by the Unique Readability Theorem:

(a) ϕ is a propositional variable.
Then define $\widetilde{v}(\phi) := v(\phi)$.

(b) $\phi = \neg\psi$. Then len(ψ) < len(ϕ). Define $\widetilde{v}(\phi)$ as follows:

ψ	$\neg\psi$
T	F
F	T

(i.e. if $\widetilde{v}(\psi) = T$ then $\widetilde{v}(\phi) := F$, and if $\widetilde{v}(\psi) = F$ then $\widetilde{v}(\phi) := T$).

(c) $\phi = (\psi \star \chi)$ where $\star$ is a binary connective. Then len(ψ) < len(ϕ) and len(χ) < len(ϕ). Define $\widetilde{v}(\phi)$ as follows:

ψ	χ	$(\psi \wedge \chi)$	$(\psi \vee \chi)$	$(\psi \to \chi)$	$(\psi \leftrightarrow \chi)$
T	T	T	T	T	T
T	F	F	T	F	F
F	T	F	T	T	F
F	F	F	F	T	T

(so e.g. if $\star$ is $\to$ and $\widetilde{v}(\psi) = T$ and $\widetilde{v}(\chi) = F$, then $\widetilde{v}(\phi) := F$).

The tables in this definition are called the **truth tables** of the connectives. They correspond to how we normally use *'not'*, *'and'*, *'or'*, *'if . . . then'*, and *'if and only if'* in mathematics (though not always to how we use them in natural language, particularly in the case of $\rightarrow$).

We can draw up more general truth tables to analyse this recursive definition of truth for more complicated formulas.

Example 3.2. "If n is prime then $n = 2$ or n is odd" is a true statement for every natural number n. To analyse this, we construct the truth table for the formula

$$\phi := (p_0 \to (p_1 \vee p_2)).$$

$\widetilde{v}(\phi)$ only depends on $v(p_0), v(p_1)$, and $v(p_2)$.

p_o	p_1	p_2	$(p_1 \vee p_2)$	ϕ
T	T	T	T	T
T	T	F	T	T
T	F	T	T	T
T	F	F	F	F
F	T	T	T	T
F	T	F	T	T
F	F	T	T	T
F	F	F	F	T

So "if n is prime then $n = 2$ or n is odd" is true unless n is prime but neither odd nor equal to 2, i.e. unless n is an even prime other than 2. But no such natural number n exists.

Example 3.3. We construct the truth table for the formula

$$\phi := ((p_0 \to p_1) \to (\neg p_1 \to \neg p_0)).$$

p_0	p_1	$(p_0 \to p_1)$	$\neg p_1$	$\neg p_0$	$(\neg p_1 \to \neg p_0)$	ϕ
T	T	T	F	F	T	T
T	F	F	T	F	F	T
F	T	T	F	T	T	T
F	F	T	T	T	T	T

3.2. Satisfaction, validity, consequences

Definition 3.4. Let ϕ be a formula.

- A valuation v **satisfies** ϕ
  if $\widetilde{v}(\phi) = T$.

- ϕ is **satisfiable**
  if ϕ is satisfied by *some* valuation.

- ϕ is **logically valid**
  if ϕ is satisfied by *every* valuation
  (e.g. Example 3.3, not Example 3.2).
  A logically valid formula is also called a
  propositional tautology.

Remark 3.5. A formula ϕ is satisfiable if and only if $\neg\phi$ is *not* logically valid.

Definition 3.6. Let Γ be any set of formulas (possibly empty, possibly infinite).

- A valuation v **satisfies** Γ if it satisfies every element of Γ.

- A formula ϕ is a **logical consequence** of Γ if every valuation satisfying Γ satisfies ϕ; i.e. if for all valuations v,

  if $\widetilde{v}(\psi) = T$ for all $\psi \in \Gamma$, then $\widetilde{v}(\phi) = T$.

  Notation: $\Gamma \vDash \phi$; "Γ entails ϕ".

Note: $\emptyset \vDash \phi$ if and only if ϕ is logically valid. We abbreviate this to $\vDash \phi$. We also often abbreviate $\{\psi\} \vDash \phi$ to $\psi \vDash \phi$.

Example 3.7.

$$\phi \vDash (\psi \to \phi).$$

Indeed, for any v with $\widetilde{v}(\phi) = T$, by tt $\to$ we have $\widetilde{v}((\psi \to \phi)) = T$

(no matter what $\widetilde{v}(\psi)$ is).

('tt $\star$' refers to the truth table of the connective $\star$)

Lemma 3.8. $\Gamma \cup \{\psi\} \models \phi$ *if and only if* $\Gamma \models (\psi \to \phi)$.

In particular, $\psi \models \phi$ *if and only if* $\models (\psi \to \phi)$.

Proof:

$\Rightarrow$: Assume $\Gamma \cup \{\psi\} \models \phi$.
Let v be any valuation which satisfies Γ.

- Case 1: $\widetilde{v}(\psi) = F$. Then $\widetilde{v}((\psi \to \phi)) = T$ by tt $\to$.
- Case 2: $\widetilde{v}(\psi) = T$. Then $\widetilde{v}$ satisfies $\Gamma \cup \{\psi\}$, so $\widetilde{v}(\phi) = T$, so then $\widetilde{v}((\psi \to \phi)) = T$ by tt $\to$.

So $\Gamma \models (\psi \to \phi)$.

$\Leftarrow$: Suppose $\Gamma \models (\psi \to \phi)$.

Let v be any valuation satisfying $\Gamma \cup \{\psi\}$.
Then $\widetilde{v}((\psi \to \phi)) = T = \widetilde{v}(\psi)$, so $\widetilde{v}(\phi) = T$ by tt $\to$.
Hence $\Gamma \cup \{\psi\} \models \phi$.

$\square$

Example 3.9. Recall from Example 3.3 that $\models ((p_0 \to p_1) \to (\neg p_1 \to \neg p_0))$. Applying Lemma 3.8 twice, we deduce first $(p_0 \to p_1) \models (\neg p_1 \to \neg p_0)$, and then $\{(p_0 \to p_1), \neg p_1\} \models \neg p_0$.

3.3. Equivalence

Definition 3.10. Two formulas ϕ, ψ are **logically equivalent**
if $\phi \vDash \psi$ and $\psi \vDash \phi$,
i.e. if $\widetilde{v}(\phi) = \widetilde{v}(\psi)$ for *every* valuation v.

Notation: $\phi \vDash\dashv \psi$

Exercise. $\phi \vDash\dashv \psi$ if and only if $\vDash (\phi \leftrightarrow \psi)$.

Lemma 3.11. *(i) For any formulas* ϕ, ψ

$$(\phi \vee \psi) \vDash\dashv \neg(\neg\phi \wedge \neg\psi).$$

(ii) Hence every formula is logically equivalent to one without $\vee$.

Proof. (i) Either use truth tables,
or observe that for any valuation v:

$$
\begin{array}{lll}
 & \widetilde{v}(\phi \vee \psi) = F & \\
\text{iff} & \widetilde{v}(\phi) = F = \widetilde{v}(\psi) & \text{by tt } \vee \\
\text{iff} & \widetilde{v}(\neg\phi) = T = \widetilde{v}(\neg\psi) & \text{by tt } \neg \\
\text{iff} & \widetilde{v}((\neg\phi \wedge \neg\psi)) = T & \text{by tt } \wedge \\
\text{iff} & \widetilde{v}(\neg(\neg\phi \wedge \neg\psi)) = F & \text{by tt } \neg
\end{array}
$$

(ii) By induction on the length of ϕ. Consider cases:

- $\phi = p_i$: clear.
- $\phi = \neg\psi$: by IH, $\psi \vDash\dashv \psi'$ for some ψ' not containing $\vee$.
  Then $\phi \vDash\dashv \neg\psi'$, which does not contain $\vee$.
- $\phi = (\psi \star \chi)$: by IH, say $\psi \vDash\dashv \psi'$ and $\chi \vDash\dashv \chi'$ where ψ' and χ' do not contain $\vee$.
  If $\star$ is not $\vee$, we conclude since $\phi \vDash\dashv (\psi' \star \chi')$.

If $\star$ is $\vee$, we conclude since
$\phi \models\!\!\!\!\dashv \neg(\neg\psi' \wedge \neg\chi')$.

$\square$

Notation 3.12. If $\phi_1, \ldots, \phi_n$ are formulas, we can write their disjunction as

$$(\ldots((\phi_1 \vee \phi_2) \vee \phi_3) \ldots \vee \phi_n).$$

This is rather cumbersome notation, so we abbreviate it to

$$\bigvee_{i=1}^{n} \phi_i.$$

Formally, we make the following recursive definitions:

$$\bigvee_{i=1}^{1} \phi_i = \phi_1 \text{ and } \bigwedge_{i=1}^{1} \phi_i = \phi_1,$$

and for $n > 1$,

$$\bigvee_{i=1}^{n} \phi_i = (\bigvee_{i=1}^{n-1} \phi_i \vee \phi_n) \text{ and } \bigwedge_{i=1}^{n} \phi_i = (\bigwedge_{i=1}^{n-1} \phi_i \wedge \phi_n).$$

So $\widetilde{v}(\bigvee_{i=1}^{n} \phi_i) = T$ iff for some i, $\widetilde{v}(\phi_i) = T$ and $\widetilde{v}(\bigwedge_{i=1}^{n} \phi_i) = T$ iff for all i, $\widetilde{v}(\phi_i) = T$.

We also sometimes write e.g. $(\phi_1 \vee \phi_2 \vee \phi_3)$ for $\bigvee_{i=1}^{3} \phi_i = ((\phi_1 \vee \phi_2) \vee \phi_3)$.

Lemma 3.13. *Let ϕ, ψ, ϕ_i be formulas. Then*

(i) $\neg(\phi \vee \psi) \models\dashv (\neg\phi \wedge \neg\psi)$
More generally,

$$\neg \bigvee_{i=1}^{n} \phi_i \models\dashv \bigwedge_{i=1}^{n} \neg\phi_i,$$

hence also

$$\neg \bigwedge_{i=1}^{n} \phi_i \models\dashv \bigvee_{i=1}^{n} \neg\phi_i.$$

These are called De Morgan's Laws.

(ii) $(\phi \to \psi) \models\dashv (\neg\phi \vee \psi)$

(iii) $(\phi \leftrightarrow \psi) \models\dashv ((\phi \to \psi) \wedge (\psi \to \phi))$

(iv) $(\phi \vee \psi) \models\dashv ((\phi \to \psi) \to \psi)$

(v) $(\phi \wedge \bigvee_{i=1}^{n} \psi_i) \models\dashv \bigvee_{i=1}^{n} (\phi \wedge \psi_i)$
("$\wedge$ distributes over $\vee$")

(vi) $(\phi \vee \bigwedge_{i=1}^{n} \psi_i) \models\dashv \bigwedge_{i=1}^{n} (\phi \vee \psi_i)$
("$\vee$ distributes over $\wedge$")

3.4. Truth functions

Definition 3.14.

- Let V_n be the set of all functions

  $$v : \{p_0, \dots, p_{n-1}\} \to \{T, F\},$$

  i.e. the "partial" valuations assigning values only to the first n propositional variables.
  Note $\#V_n = 2^n$.

- An n-**ary truth function** is a function

  $$J : V_n \to \{T, F\}.$$

  There are precisely 2^{2^n} such functions.

- Let $\mathrm{Form}_n(\mathcal{L}_{\mathrm{prop}})$ be the set of formulas which contain only propositional variables from the set $\{p_0, \dots, p_{n-1}\}$.

  If $\phi \in \mathrm{Form}_n(\mathcal{L}_{\mathrm{prop}})$ and $v \in V_n$, then $\widetilde{v}(\phi)$ is well-defined, so ϕ **represents** an n-ary truth function

$$J^n_\phi : V_n \to \{T, F\};\ v \mapsto \widetilde{v}(\phi).$$

Remark 3.15. Formulas $\phi, \psi \in \mathrm{Form}_n(\mathcal{L}_{\text{prop}})$ are logically equivalent if and only if the same valuations satisfy them, so

$$\phi \vDash\dashv \psi \ \Leftrightarrow\ J^n_\phi = J^n_\psi.$$

In other words, a formula in $\mathrm{Form}_n(\mathcal{L}_{\text{prop}})$ is determined up to logical equivalence by the n-ary truth function it represents.

Definition 3.16. A formula is in **disjunctive normal form** (**DNF**) if it is of the form

$$\bigvee_{i=1}^{k} \bigwedge_{j=1}^{s_i} \psi_{i,j}$$

where each $\psi_{i,j}$ is either a propositional variable or the negation of a propositional variable.

Example 3.17.
$(((p_1 \wedge \neg p_2) \vee p_0) \vee ((\neg p_0 \wedge \neg p_3) \wedge p_0))$ is in disjunctive normal form. So are p_2 and $\neg p_7$.

Theorem 3.18.
For every $n \geq 1$, every n-ary truth function $J : V_n \to \{T, F\}$, is represented by a formula in disjunctive normal form.

In particular, every formula is logically equivalent to one in DNF.

Proof: Let

$$U := \{v \in V_n \mid J(v) = T\}.$$

First, suppose $k := |U| > 0$, say $U = \{v_0, ..., v_{k-1}\}$. Set

$$\phi := \bigvee_{i=0}^{k-1} \bigwedge_{j=0}^{n-1} \psi_{i,j}$$

where

$$\psi_{i,j} := \begin{cases} p_j & \text{if} \quad v_i(p_j) = T \\ \neg p_j & \text{if} \quad v_i(p_j) = F. \end{cases}$$

Then for any $w \in V_n$ and $i < k$,

$$\widetilde{w}(\bigwedge_{j=0}^{n-1} \psi_{i,j}) = T \;\Leftrightarrow\; w = v_i.$$

Hence for any $w \in V_n$, we have

$$\begin{aligned} J_\phi^n(w) = T \;&\Leftrightarrow\; \widetilde{w}(\phi) = T \\ &\Leftrightarrow\; (w = v_0 \text{ or } \ldots \text{ or } w = v_{k-1}) \\ &\Leftrightarrow\; w \in U \\ &\Leftrightarrow\; J(w) = T, \end{aligned}$$

so $J_\phi^n = J$, i.e. ϕ represents J.

Finally, we handle the special case that $U = \emptyset$, i.e. $J(v) = F$ for all $v \in V_n$, by setting $\phi := (p_0 \wedge \neg p_0)$ (which is in DNF). Then ϕ represents J, since $J_\phi^n(v) = \widetilde{v}(\phi) = F = J(v)$ for any $v \in V_n$.

$\square$

Definition 3.19.

- If $*_1, ..., *_k$ are truth-functional connectives with associated truth tables (unary, binary, or even ternary or higher), write $\mathcal{L}_{\text{prop}}[*_1, ..., *_k]$ for the language with these connectives instead of $\neg, \rightarrow, \wedge, \vee, \leftrightarrow$, and define $\text{Form}(\mathcal{L}_{\text{prop}}[*_1, ..., *_k])$ and $\text{Form}_n(\mathcal{L}_{\text{prop}}[*_1, ..., *_k])$ accordingly.

- Say $\mathcal{L}_{\text{prop}}[*_1, ..., *_k]$ is **adequate** if every n-ary truth function (for $n \geq 1$) is represented by some $\phi \in \text{Form}_n(\mathcal{L}_{\text{prop}}[*_1, ..., *_k])$.

Lemma 3.20. *The following languages are adequate:*

(i) $\mathcal{L}_{\text{prop}}[\neg, \wedge, \vee]$.

(ii) $\mathcal{L}_{\text{prop}}[\neg, \wedge]$.

(iii) $\mathcal{L}_{\text{prop}}[\neg, \vee]$.

(iv) $\mathcal{L}_{\text{prop}}[\neg, \to]$.

Proof.

(i) Theorem 3.18.

(ii) By (i) and De Morgan's law

$$(\phi \vee \psi) \models\!\!\!\!\dashv \neg(\neg\phi \wedge \neg\psi),$$

via the argument of Lemma 3.11.

(iii) Similarly, using De Morgan's other law

$$(\phi \wedge \psi) \models\!\!\!\!\dashv \neg(\neg\phi \vee \neg\psi).$$

(iv) Similarly, using (iii) and the equivalence (Lemma 3.13(iv))

$$(\phi \vee \psi) \vDash\Dashv ((\phi \to \psi) \to \psi).$$

$\square$

Remark 3.21.

- $\mathcal{L} := \mathcal{L}_{\text{prop}}[\vee, \wedge, \to]$ is **not** adequate: defining v_T by $v_T(p_i) = T$ for all i, we have $v_T(\phi) = T$ for all $\phi \in \text{Form}(\mathcal{L}$, so in particular no such ϕ gives $J^1_\phi = J^1_{\neg p_0}$.

- There are precisely two binary connectives, say $\uparrow$ and $\downarrow$, such that $\mathcal{L}_{\text{prop}}[\uparrow]$ and $\mathcal{L}_{\text{prop}}[\downarrow]$ are adequate.

4. Proofs

- We introduced '*logical consequence*' – $\Gamma \models \phi$ means: whenever (each formula of) Γ is true, so is ϕ.

- If Γ is finite, we can check whether $\Gamma \models \phi$ by considering truth tables. But for infinite Γ, it is less clear how to determine when $\Gamma \models \phi$ holds.

- We now define a notion of a **proof** of a formula ϕ from hypotheses Γ, and we will show that $\Gamma \models \phi$ if and only if such a proof exists ("completeness").

- Generally, a proof system can be defined by choosing some **axioms** and some **rules of inference**. Then a **proof** of ϕ from Γ is a finite sequence
  $\phi_1, \phi_2, \ldots, \phi_n$ such that

$\phi_n = \phi$, and for each $i = 1, \ldots, n$:

- $\phi_i \in \Gamma$,
- or ϕ_i is some **axiom**,
- or ϕ_i follows from previous ϕ_j's by a rule of inference.

We work with $\mathcal{L}_0 := \mathcal{L}_{\text{prop}}[\neg, \rightarrow]$. Since $\mathcal{L}_0$ is adequate by Lemma 3.20(iv), we lose nothing by considering only $\mathcal{L}_0$.

Definition 4.1.

The **proof system** L_0 consists of the following axioms and rules:

Axioms

An **axiom** of L_0 is any formula of the following form, where $\alpha, \beta, \gamma \in \text{Form}(\mathcal{L}_0)$:

A1 $(\alpha \rightarrow (\beta \rightarrow \alpha))$

A2 $((\alpha \rightarrow (\beta \rightarrow \gamma)) \rightarrow ((\alpha \rightarrow \beta) \rightarrow (\alpha \rightarrow \gamma)))$

A3 $((\neg\alpha \rightarrow \beta) \rightarrow ((\neg\alpha \rightarrow \neg\beta) \rightarrow \alpha))$

Rules of inference

Just one rule, **modus ponens**:

MP For any $\phi, \psi \in \text{Form}(\mathcal{L}_0)$:
From ϕ and $(\phi \rightarrow \psi)$, infer ψ.

Definition 4.2. Let $\Gamma \subseteq \text{Form}(\mathcal{L}_0)$. A formula $\phi \in \text{Form}(\mathcal{L}_0)$ is **provable** from **hypotheses** Γ, written

$$\Gamma \vdash \phi,$$

if there is a sequence of $\mathcal{L}$-formulas (a **derivation** or **proof**) $\phi_1, \ldots, \phi_n$ with $\phi_n = \phi$ such that for each $i \leq n$, at least one of the following holds:

- (A1-A3) ϕ_i is an axiom.
- (Hyp) $\phi_i \in \Gamma$.
- (MP) $\phi_k = (\phi_j \to \phi_i)$ for some $j, k < i$.

In the case $\Gamma = \emptyset$, we usually write $\vdash \phi$ rather than $\emptyset \vdash \phi$, and we say that ϕ is a **theorem** of the system L_0.

Note that if $\Delta \vdash \phi$ and $\Delta' \supseteq \Delta$, then also $\Delta' \vdash \phi$.

The term **propositional calculus** is sometimes used to refer to L_0 or similar proof systems. It is also sometimes used to refer to propositional logic in general.

Example 4.3. For any $\phi \in \mathrm{Form}(\mathcal{L}_0)$

$$\vdash (\phi \to \phi).$$

Proof:

1	$((\phi \to ((p_0 \to \phi) \to \phi)) \to ((\phi \to (p_0 \to \phi)) \to (\phi \to \phi)))$	[A2 with $\alpha = \phi,$
2	$(\phi \to ((p_0 \to \phi) \to \phi))$	[A1 with $\alpha = \phi,$
3	$((\phi \to (p_0 \to \phi)) \to (\phi \to \phi))$	[MP $2, 1$]
4	$(\phi \to (p_0 \to \phi))$	[A1 with $\alpha = \phi,$
5	$(\phi \to \phi)$	[MP $4, 3$]

Then this sequence of formulas is a proof of $(\phi \to \phi)$ from $\emptyset$ in L_0.

□

Example 4.4.
For any $\phi, \psi \in \text{Form}(\mathcal{L}_0)$:

$$\{\psi, \neg\psi\} \vdash \phi$$

Proof:

1	$((\neg\phi \to \psi) \to ((\neg\phi \to \neg\psi) \to \phi))$	[A3]
2	ψ	[Hyp]
3	$(\psi \to (\neg\phi \to \psi))$	[A1]
4	$(\neg\phi \to \psi)$	[MP $2, 3$]
5	$((\neg\phi \to \neg\psi) \to \phi)$	[MP $4, 1$]
6	$\neg\psi$	[Hyp]
7	$(\psi \to (\neg\phi \to \neg\psi))$	[A1]
8	$(\neg\phi \to \neg\psi)$	[MP $6, 7$]
9	ϕ	[MP $8, 5$]

□

Theorem 4.5 (The Soundness Theorem for L_0). *L_0 is* **sound**, *i.e. for any* $\Gamma \subseteq \text{Form}(\mathcal{L}_0)$ *and for any* $\phi \in \text{Form}(\mathcal{L}_0)$*:*

$$\textit{If } \Gamma \vdash \phi \textit{ then } \Gamma \vDash \phi.$$

In particular, any theorem of L_0 is logically valid.

Proof:
We show by (complete) induction on m:

($\star$) If a formula ϕ has a proof of length m from Γ in L_0, then $\Gamma \vDash \phi$.

So suppose $\alpha_1, \ldots, \alpha_m$ is a proof in L_0, and ($\star$) holds for all $m' < m$. We have to show that $\Gamma \vDash \alpha_m$.

Case 1: α_m is an axiom.
One verifies by truth tables (exercise) that our axioms are logically valid, so $\Gamma \models \alpha_m$.

Case 2: $\alpha_m \in \Gamma$.
Then $\Gamma \models \alpha_m$.

Case 3: α_m is obtained by MP.
So say $j, k < m$ and $\alpha_k = (\alpha_j \to \alpha_m)$.

By the inductive hypothesis,
since $\alpha_1, \ldots, \alpha_j$ is a proof of length $j < m$,
we have $\Gamma \models \alpha_j$.
Similarly $\Gamma \models \alpha_k$, i.e. $\Gamma \models (\alpha_j \to \alpha_m)$.

But $\{\alpha_j, (\alpha_j \to \alpha_m)\} \models \alpha_m$ by Lemma 3.8,
and it follows (from the definition of $\models$) that
$\Gamma \models \alpha_m$.

$\square$

4.1. The Deduction Theorem for L_0

A common pattern of reasoning goes as follows: "Suppose A holds. Then [some chain of reasoning], and so B holds. Hence A implies B." The Deduction Theorem implements this.

Theorem 4.6 (The Deduction Theorem for L_0). *For any* $\Gamma \subseteq \text{Form}(\mathcal{L}_0)$ *and for any* $\phi, \psi \in \text{Form}(\mathcal{L}_0)$,

$$\text{if } \Gamma \cup \{\phi\} \vdash \psi \text{ then } \Gamma \vdash (\phi \to \psi).$$

Proof: We prove this by induction on the length of a proof of ψ from $\Gamma \cup \{\phi\}$.

So suppose $\alpha_1, \ldots, \alpha_m$ is a proof in L_0 from $\Gamma \cup \{\phi\}$,
and we show $\Gamma \vdash (\phi \to \alpha_m)$,
assuming inductively that $\Gamma \vdash (\phi \to \alpha_i)$ for all $i < m$.

Case 1: α_m is an axiom.
Then $\Gamma \vdash (\phi \to \alpha_m)$:

1	α_m	[A1/2/3]
2	$(\alpha_m \to (\phi \to \alpha_m))$	[A1]
3	$(\phi \to \alpha_m)$	[MP 1,2]

Case 2: $\alpha_m \in \Gamma \cup \{\phi\}$.
If $\alpha_m \in \Gamma$ then the proof above works (changing the justification on line 1 to "[Hyp]"). Otherwise $\alpha_m = \phi$, and then $\vdash (\phi \to \alpha_m)$ by Example 4.3, and hence $\Gamma \vdash (\phi \to \alpha_m)$.

Case 3: α_m is obtained by MP from some earlier α_j, α_k, i.e. there are $j, k < m$ such that $\alpha_j = (\alpha_k \to \alpha_m)$.

By the induction hypothesis, we have

$$\begin{aligned} & \Gamma \vdash (\phi \to \alpha_k) \\ \text{and} \quad & \Gamma \vdash (\phi \to (\alpha_k \to \alpha_m)) \end{aligned}$$

So say

$$\beta_1, \ldots, \beta_{r-1}, (\phi \to \alpha_k)$$

and

$$\gamma_1, \ldots, \gamma_{s-1}, (\phi \to (\alpha_k \to \alpha_m))$$

are proofs in L_0 from Γ.

Then

1	β_1	
⋮	⋮	
r-1	β_{r-1}	
r	$(\phi \to \alpha_k)$	
r+1	γ_1	
⋮	⋮	
r+s-1	γ_{s-1}	
r+s	$(\phi \to (\alpha_k \to \alpha_m))$	
r+s+1	$((\phi \to (\alpha_k \to \alpha_m)) \to$ $((\phi \to \alpha_k) \to (\phi \to \alpha_m)))$	[A2]
r+s+2	$((\phi \to \alpha_k) \to (\phi \to \alpha_m))$	[MP r+s, r+s+1]
r+s+3	$(\phi \to \alpha_m)$	[MP r, r+s+2]

is a proof of $(\phi \to \alpha_m)$ in L_0 from Γ.

□

Remark 4.7.

- The proof only used instances of A1, A2, and the rule MP.

- The proof gives a precise **algorithm** for converting any proof showing $\Gamma \cup \{\phi\} \vdash \psi$ into one showing $\Gamma \vdash (\phi \to \psi)$.

- The converse implication is immediate from MP:

If $\Gamma \vdash (\phi \to \psi)$ then $\Gamma \cup \{\phi\} \vdash \psi$:

$\vdots$	$\vdots$	proof from Γ
r	$(\phi \to \psi)$	
r+1	ϕ	[Hyp]
r+2	ψ	[MP r, r+1]

More generally:

Remark 4.8. If $\Gamma \vdash (\phi \to \psi)$ and $\Gamma \vdash \phi$, then $\Gamma \vdash \psi$ by MP.

Explicitly: if $\alpha_1, \ldots, \alpha_{n-1}, (\phi \to \psi)$ and $\beta_1, \ldots, \beta_{m-1}, \phi$ are proofs from Γ, then so is $\alpha_1, \ldots \alpha_{n-1}, (\phi \to \psi), \beta_1, \ldots, \beta_{m-1}, \phi, \psi$.

Example 4.9. If $\Gamma \vdash (\phi \to \psi)$ and $\Gamma \vdash (\psi \to \chi)$, then $\Gamma \vdash (\phi \to \chi)$.

Proof: By the deduction theorem, it suffices to show that $\Gamma \cup \{\phi\} \vdash \chi$.

Now $\Gamma \cup \{\phi\} \vdash (\phi \to \psi)$ and $\Gamma \cup \{\phi\} \vdash \phi$, so $\Gamma \cup \{\phi\} \vdash \psi$ by MP (Remark 4.8).

Then since $\Gamma \cup \{\phi\} \vdash (\psi \to \chi)$, we have $\Gamma \cup \{\phi\} \vdash \chi$ by MP again.

$\square$

Lemma 4.10. *If* $\Gamma \cup \{\phi\} \vdash \psi$ *and* $\Gamma \vdash \phi$*, then* $\Gamma \vdash \psi$*.*

Proof: We have $\Gamma \vdash (\phi \to \psi)$ by the deduction theorem, so $\Gamma \vdash \psi$ by MP.

(Alternative direct argument: if $\alpha_1, \ldots, \alpha_{n-1}, \phi$ is a proof from Γ and $\beta_1, \ldots, \beta_{m-1}, \psi$ is a proof from $\Gamma \cup \{\phi\}$, then $\alpha_1, \ldots, \alpha_{n-1}, \beta_1, \ldots, \beta_{m-1}, \psi$ is a proof from Γ.)

$\square$

5. Completeness and Compactness

Theorem 5.1 (The Completeness Theorem for L_0)**.**
L_0 is **complete***, i.e. for any* $\Gamma \subseteq \mathrm{Form}(\mathcal{L}_0)$ *and for any* $\phi \in \mathrm{Form}(\mathcal{L}_0)$*:*

$$\textit{If } \Gamma \models \phi \textit{ then } \Gamma \vdash \phi.$$

Given also soundness, it follows:
$\Gamma \models \phi$ iff $\Gamma \vdash \phi$.

To prove completeness, it is convenient to go via a proof-theoretic analogue of satisfiability called *consistency*.

Definition 5.2.
$\Gamma \subseteq \mathrm{Form}(\mathcal{L}_0)$ is **inconsistent**
if for some $\chi \in \mathrm{Form}(\mathcal{L}_0)$,

$$\Gamma \vdash \chi \text{ and } \Gamma \vdash \neg\chi.$$

Otherwise, Γ is **consistent**.

Lemma 5.3. *Any satisfiable* $\Gamma \subseteq \text{Form}(\mathcal{L}_0)$ *is consistent.*

Proof. Suppose Γ is inconsistent, say $\Gamma \vdash \chi$ and $\Gamma \vdash \neg\chi$.
Then $\Gamma \models \chi$ and $\Gamma \models \neg\chi$ by soundness.
But no valuation satisfies both χ and $\neg\chi$, so Γ is not satisfiable. □

Lemma 5.4.

(i) $\Gamma \vdash \phi$ *if and only if* $\Gamma \cup \{\neg\phi\}$ *is inconsistent.*

(ii) $\Gamma \models \phi$ *if and only if* $\Gamma \cup \{\neg\phi\}$ *is unsatisfiable.*

Proof:

(i) Suppose $\Gamma \cup \{\neg\phi\}$ is inconsistent, say $\Gamma \cup \{\neg\phi\} \vdash \chi$ and $\Gamma \cup \{\neg\phi\} \vdash \neg\chi$.

Then by the deduction theorem, $\Gamma \vdash (\neg\phi \to \chi)$ and $\Gamma \vdash (\neg\phi \to \neg\chi)$.

But

$$((\neg\phi \to \chi) \to ((\neg\phi \to \neg\chi) \to \phi))$$

is an instance of A3, so by MP twice, we conclude $\Gamma \vdash \phi$.

Conversely, if $\Gamma \vdash \phi$ then $\Gamma \cup \{\neg\phi\}$ is inconsistent, since $\Gamma \cup \{\neg\phi\} \vdash \phi$ and $\Gamma \cup \{\neg\phi\} \vdash \neg\phi$.

(ii) $\Gamma \models \phi \Leftrightarrow$ any valuation satisfying Γ satisfies ϕ
$\Leftrightarrow$ no valuation satisfying Γ satisfies $\neg\phi$
$\Leftrightarrow \Gamma \cup \{\neg\phi\}$ is unsatisfiable.

$\square$

So to prove the Completeness Theorem, it suffices to prove that any consistent Γ is satisfiable.

Definition 5.5.
$\Gamma \subseteq \text{Form}(\mathcal{L}_0)$ is **complete** (or *maximal consistent*) if

- Γ is consistent, and
- for *every* $\phi \in \text{Form}(\mathcal{L}_0)$, either $\Gamma \vdash \phi$ or $\Gamma \vdash \neg\phi$.

Warning. This notion of completeness of a set of formulas is quite distinct from the notion of completeness of a proof system! In the Completeness Theorem we are proving, as well as Gödel's Completeness Theorem for first-order logic which we will prove later, "completeness" refers to completeness of a proof system ("$\models \Rightarrow \vdash$"). In Gödel's Incompleteness Theorems (the subject of a Part C course) meanwhile, "completeness" refers to completeness of a set of (first-order) formulas, as in Definition 5.5.

We will prove the completeness theorem by first showing that every consistent Γ extends to a complete set, then showing that complete sets are satisfiable.

Lemma 5.6.
If $\Gamma \subseteq \text{Form}(\mathcal{L}_0)$ is consistent and $\phi \in \text{Form}(\mathcal{L}_0)$, then either $\Gamma \cup \{\phi\}$ is consistent or $\Gamma \cup \{\neg\phi\}$ is consistent.

Proof: If $\Gamma \nvdash \phi$, then $\Gamma \cup \{\neg\phi\}$ is consistent by Lemma 5.4(i). Otherwise, $\Gamma \vdash \phi$. Then $\Gamma \cup \{\phi\}$ is consistent, since otherwise for some χ we have $\Gamma \cup \{\phi\} \vdash \chi$ and $\Gamma \cup \{\phi\} \vdash \neg\chi$, and hence $\Gamma \vdash \chi$ and $\Gamma \vdash \neg\chi$ (by Lemma 4.10), contradicting consistency of Γ.

$\square$

Theorem 5.7.
Suppose Γ is consistent. Then there is a complete $\Gamma' \supseteq \Gamma$.

Proof:
Form($\mathcal{L}_0$) is countable (Fact 2.9), so say

$$\text{Form}(\mathcal{L}_0) = \{\phi_0, \phi_1, \phi_2, \ldots\}.$$

Construct a chain of consistent sets

$$\Gamma = \Gamma_0 \subseteq \Gamma_1 \subseteq \Gamma_2 \subseteq \ldots$$

as follows:

- $\Gamma_0 := \Gamma$.
- Given consistent Γ_n, let

  $$\Gamma_{n+1} := \begin{cases} \Gamma_n \cup \{\phi_n\} & \text{if } \Gamma_n \cup \{\phi_n\} \text{ is consistent} \\ \Gamma_n \cup \{\neg\phi_n\} & \text{otherwise} \end{cases}$$

  Then Γ_{n+1} is consistent by Lemma 5.6.

Now let $\Gamma' := \bigcup_{n=0}^{\infty} \Gamma_n$.
Then Γ' is consistent:
if $\Gamma' \vdash \chi$ and $\Gamma' \vdash \neg\chi$, then proofs witnessing this use only finitely many formulas from Γ' as hypotheses, so for some n, $\Gamma_n \vdash \chi$ and $\Gamma_n \vdash \neg\chi$, contradicting the consistency of Γ_n.

Finally, Γ' is complete: for all n, either $\phi_n \in \Gamma'$ or $\neg\phi_n \in \Gamma'$,
so in particular either $\Gamma' \vdash \phi_n$ or $\Gamma' \vdash \neg\phi_n$.

$\square$

Lemma 5.8.
Suppose Γ *is complete.*
Then for every $\psi, \chi \in \text{Form}(\mathcal{L}_0)$*:*

(a) $\Gamma \vdash \neg\psi$ *iff* $\Gamma \nvdash \psi$*.*
(b) $\Gamma \vdash (\psi \to \chi)$ *iff either* $\Gamma \nvdash \psi$ *or* $\Gamma \vdash \chi$*.*

Proof:

(a) Immediate from the definition of Γ being complete.

(b) **'$\Rightarrow$':** By MP, if $\Gamma \vdash (\psi \to \chi)$ and $\Gamma \vdash \psi$, then $\Gamma \vdash \chi$.

'$\Leftarrow$': Suppose $\Gamma \nvdash \psi$. Then $\Gamma \vdash \neg\psi$ by (a). But $\Gamma \cup \{\psi, \neg\psi\} \vdash \chi$ by Example 4.4, so then $\Gamma \cup \{\neg\psi\} \vdash (\psi \to \chi)$ by the deduction theorem, and so $\Gamma \vdash (\psi \to \chi)$ (by Lemma 4.10).

If $\Gamma \vdash \chi$ then $\Gamma \vdash (\psi \to \chi)$ by A1 and MP.

□

Theorem 5.9.
Suppose Γ *is complete.*
Then Γ *is satisfiable.*

Proof:

Define a valuation v by

$$v(p_i) = T \text{ iff } \Gamma \vdash p_i.$$

We prove by induction on ϕ:
Claim. *For all* $\phi \in \text{Form}(\mathcal{L}_0)$,
$\widetilde{v}(\phi) = T$ *iff* $\Gamma \vdash \phi$.

Case 1: $\phi = p_i$ for some i. Then we are done by the definition of v.

Case 2: $\phi = \neg\psi$. Then

$$\begin{array}{llll} \tilde{v}(\phi) = T & \text{iff} & \tilde{v}(\psi) = F & \text{tt } \neg \\ & \text{iff} & \Gamma \nvdash \psi & \text{IH} \\ & \text{iff} & \Gamma \vdash \neg\psi & \text{Lemma 5.8(a)} \\ & \text{iff} & \Gamma \vdash \phi & \end{array}$$

Case 3: $\phi = (\psi \to \chi)$. Then

$$\begin{array}{llll} \tilde{v}(\phi) = T & \text{iff} & \tilde{v}(\psi) = F \text{ or } \tilde{v}(\chi) = T & \text{tt } \to \\ & \text{iff} & \Gamma \nvdash \psi \text{ or } \Gamma \vdash \chi & \text{IH} \\ & \text{iff} & \Gamma \vdash (\psi \to \chi) & \text{Lemma 5.8(b)} \\ & \text{iff} & \Gamma \vdash \phi & \end{array}$$

The Claim is proven, so in particular $\tilde{v}(\phi) = T$ for all $\phi \in \Gamma$, i.e. v satisfies Γ.

$\square$

Theorem 5.10. *Let* $\Gamma \subseteq \text{Form}(\mathcal{L}_0)$*. Then* Γ *is consistent if and only if it is satisfiable.*

Proof: If Γ is consistent,
then by Theorem 5.7 it extends to a complete set,
which by Theorem 5.9 is satisfiable,
hence Γ is also satisfiable.

The converse is Lemma 5.3.

□

Proof: [Proof of Completeness Theorem 5.1]
Immediate from Lemma 5.4 and Theorem 5.10.

□

Theorem 5.11 (The Compactness Theorem for $\mathcal{L}_0$).
$\Gamma \subseteq \text{Form}(\mathcal{L}_0)$ *is satisfiable iff every finite subset of* Γ *is satisfiable.*

Proof: By Theorem 5.10, this is equivalent to:
$\Gamma \subseteq \text{Form}(\mathcal{L}_0)$ *is consistent iff every finite subset of* Γ *is consistent.*

But indeed, by finiteness of proofs,
$\Gamma \vdash \chi$ and $\Gamma \vdash \neg\chi$ iff already
$\Gamma_0 \vdash \chi$ and $\Gamma_0 \vdash \neg\chi$ for some finite $\Gamma_0 \subseteq \Gamma$.

□

Remark 5.12. Our proof of completeness used that the language was countable (in the proof of Theorem 5.7). One could also consider uncountable languages, for example with a propositional variable p_r for each real number r. Completeness can then be proven along the same lines, but it requires some form of the Axiom of Choice*.

*Namely, it is equivalent modulo ZF to the Boolean

Part II: First-order Logic

Consider: "If everyone loves their mother, then everyone loves someone."

Propositional logic can treat only the implication, $(p_0 \to p_1)$. We now introduce a refinement of propositional logic, known as *first-order* or *predicate* logic, which can capture the full meaning – rendering it as

$$(\forall x_0 L(x_0, m(x_0))) \to \forall x_0 \exists x_1 L(x_0, x_1))$$

where L is a "binary relation symbol" interpreted as loving, and m is a "unary function symbol" interpreted as $m(x)$ being the mother of x.

First-order logic extends propositional logic with universal and existential quantifiers,

Prime Ideal Theorem; see section 2.3 in Jech's book "The Axiom of Choice" for a proof.

predicates and relations, and functions and constants. The result is highly constrained compared to natural language, but is just expressive enough for the purpose of formalising mathematical statements.

As in the propositional case, we first formally define the syntax, then give a precise definition of truth, then proceed to find a sound and complete proof calculus. We will then consider compactness and other consequences for actual mathematical structures.

6. Syntax

A **countable first-order language** $\mathcal{L}$ consists of a countable set of **non-logical symbols**, along with a categorisation of its elements as each being of exactly one of the following kinds:

- A k-ary *predicate symbol*[*] for some $k \geq 1$.
- A k-ary *function symbol* for some $k \geq 1$.
- A *constant symbol*.

The alphabet of $\mathcal{L}$ consists of its non-logical symbols along with the following disjoint set of **logical symbols**:

- *Connectives:* $\rightarrow, \neg$
- *Quantifier:* $\forall$ ('for all')
- *Variables:* $x_0, x_1, x_2, \ldots$ (one variable x_i for each $i \in \mathbb{N}$)
- *3 punctuation marks:* , ( )
- *Equality symbol:* $\doteq$

[*]We often say "unary", "binary", "ternary" for "k-ary" with $k = 1, 2, 3$. When $k \geq 2$, a k-ary predicate symbol is also often called a k-ary *relation* symbol.

We recursively define *terms* and *formulas*:

Definition 6.1.

(a) A string is an $\mathcal{L}$-**term** if it has one of the following forms:
   (i) A variable x_i.
   (ii) A constant symbol.
   (iii) $f(t_1, \ldots, t_k)$ where f is a k-ary function symbol in $\mathcal{L}$ and $t_1, \ldots, t_k$ are terms.

(b) An **atomic** $\mathcal{L}$-**formula** is any string of the form

$$P(t_1, \ldots, t_k) \text{ or } t_1 \doteq t_2$$

where $k \geq 1$, $P \in \mathcal{L}$ is a k-ary relation symbol in $\mathcal{L}$, and all t_i are $\mathcal{L}$-terms.

(c) A string is a $\mathcal{L}$-**formula** if it has one of the following forms:
   (i) An atomic $\mathcal{L}$-formula.
   (ii) $\neg\phi$ or $(\phi \to \psi)$ where ϕ, ψ are $\mathcal{L}$-formulas.
   (iii) $\forall x_i \phi$ where ϕ is an $\mathcal{L}$-formula and $i \in \mathbb{N}$.

Example 6.2. The most general countable first-order language has a countably infinite set of symbols of each type:

$$\mathcal{L}_{\text{pred}} := \{(P_i^{(k)})_{i,k>0}, (f_i^{(k)})_{i,k>0}, (c_i)_{i>0}\},$$

where each $P_i^{(k)}$ is a k-ary predicate symbol, each $f_i^{(k)}$ is a k-ary function symbol, and each c_i is a constant symbol.

- The following are all $\mathcal{L}_{\text{pred}}$-terms:

  $$c_3, \quad x_5, \quad f_3^{(1)}(c_2), \quad f_1^{(2)}(x_1, f_1^{(1)}(c_{37})).$$

- $f_2^{(3)}(x_1, x_2)$ is *not* a term (wrong arity).
- $P_2^{(3)}(x_4, c_2, f_3^{(2)}(c_1, x_2))$ and $f_1^{(2)}(c_5, x_2) \doteq x_3$ are atomic formulas.
- $\forall x_1 f_2^{(2)}(x_1, c_7) \doteq x_2$ and $(\forall x_2 (P_1^{(1)}(x_3) \wedge P_1^{(2)}(x_4, x_3)) \rightarrow c_1 \doteq c_0)$ are non-atomic formulas.

Example 6.3. A more typical example of a first-order language appearing in mathematics is the *language of ordered rings*

$$\mathcal{L}_{\text{o.ring}} := \{<, \cdot, +, -, 0, 1\},$$

where $<$ is a binary relation symbol, $\cdot$, $+$, and $-$ are binary function symbols, and 0 and 1 are constant symbols. (Note that we are using these symbols as abstract symbols – forget for now the meanings we usually give to them.)

When dealing with binary function and relation symbols, we often allow ourselves to use "infix notation" as an abbreviation, so e.g.

$$\forall x_0 x_0 < x_0 + 1$$

abbreviates the $\mathcal{L}_{\text{o.ring}}$-formula

$$\forall x_0 <(x_0, +(x_0, 1)).$$

Exercise 6.4. We have **unique readability** for terms, for atomic formulas, and for formulas.

As in Fact 2.9, we have

Fact 6.5. *For any given countable first-order language $\mathcal{L}$, the sets* $\mathrm{Term}(\mathcal{L})$ *of terms in $\mathcal{L}$ and* $\mathrm{Form}(\mathcal{L})$ *of formulas in $\mathcal{L}$ are countable.*

From now on, we consider only countable first-order languages, so we often refer to them just as "languages".

7. Semantics

7.1. Informal discussion

The truth value of a propositional formula is determined by the truth values of the propositional variables. We now consider what information we need to determine the truth value of a first-order formula.

To determine the truth of

$$\phi := \forall x_0 (P(x_0) \to P(f(x_0)))$$

we first need to decide the *domain* of quantification: a non-empty set M. We then read $\forall x_0$ as "for all $x_0 \in M$".

Then, for each possible *assignment* of x_0 to an element of M, we want to determine whether

$$\psi := (P(x_0) \to P(f(x_0)))$$

holds.

For this, we need:

- an interpretation of f as a function $M \to M$, so that $f(x_0)$ denotes an element of M;

- an interpretation of P as a choice of True/False for each element of M (i.e. as a a subset of M), so that $P(x_0)$ and $P(f(x_0))$ are given truth values.

For example, taking $M = \mathbb{Z}$, interpreting f as the successor function $S(n) = n + 1$ and P as the natural numbers $\mathbb{N} \subseteq \mathbb{Z}$, ϕ is true because ψ is true for any assignment of x_0, but ϕ is false if we interpret P as $-\mathbb{N}$, because ψ is false when we assign x_0 to $0 \in \mathbb{Z}$.

So in general, to evaluate the truth of a formula we need:

- a domain;
- interpretations of the non-logical symbols;
- an assignment of the variables to elements of the domain.

7.2. Interpretations and Assignments

Definition 7.1.
Let $\mathcal{L}$ be a language. An $\mathcal{L}$-**structure** $\mathcal{M}$ consists of:

- A non-empty set M, the **domain** of $\mathcal{M}$;
- For each k-ary function symbol $f \in \mathcal{L}$, a k-ary function $f^{\mathcal{M}} : M^k \to M$;
- For each k-ary predicate symbol $P \in \mathcal{L}$, a subset $P^{\mathcal{M}} \subseteq M^k$;
- For each constant symbol $c \in \mathcal{L}$, an element $c^{\mathcal{M}} \in M$.

An **interpretation** of a language $\mathcal{L}$ is precisely a choice of an $\mathcal{L}$-structure.

Notation 7.2. Consider for example the language $\mathcal{L} = \{f, P, c\}$ with f a binary function symbol, P a unary predicate symbol, and c a constant symbol.

We denote an $\mathcal{L}$-structure by $\mathcal{M} = \langle M; f^{\mathcal{M}}, P^{\mathcal{M}}, c^{\mathcal{M}} \rangle$.

We also write e.g. $\mathcal{M} = \langle \mathbb{N}; +, 2\mathbb{N}, 3 \rangle$ for the $\mathcal{L}$-structure with domain $\mathbb{N}$ and with f interpreted as the addition function $f^{\mathcal{M}} : \mathbb{N}^2 \to \mathbb{N}; (x, y) \mapsto x + y$, P as the subset $P^{\mathcal{M}} = 2\mathbb{N} \subseteq \mathbb{N}$, and c as the element $c^{\mathcal{M}} = 3 \in \mathbb{N}$.

Definition 7.3. Let $\mathcal{M} = \langle M; \ldots \rangle$ be an $\mathcal{L}$-structure with domain M.

- An **assignment** in $\mathcal{M}$ is a function

$$a : \{x_0, x_1, \ldots\} \to M$$

- An assignment a extends to a function

$$\widetilde{a} : \text{Terms}(\mathcal{L}) \to M$$

  defined recursively as follows:

  – $\widetilde{a}(x_i) := a(x_i)$ where $i \in \mathbb{N}$.

  – $\widetilde{a}(c) := c^{\mathcal{M}}$ where $c \in \mathcal{L}$ is a constant symbol.

  – $\widetilde{a}(f(t_1, \ldots, t_k)) := f^{\mathcal{M}}(\widetilde{a}(t_1), \ldots, \widetilde{a}(t_k))$ where $f \in \mathcal{L}$ is a k-ary function symbol and $t_i \in \text{Term}(\mathcal{L})$.

Given an assignment a in $\mathcal{M}$, we recursively define whether

$$\mathcal{M} \models_a \phi$$

(read as "*ϕ holds in $\mathcal{M}$ under the assignment a*", or "*$\mathcal{M}$ satisfies ϕ under a*"; sometimes also written as $\mathcal{M} \models \phi[a]$), as follows:

- $\mathcal{M} \models_a P(t_1, \ldots, t_k)$ if and only if $(\widetilde{a}(t_1), \ldots, \widetilde{a}(t_k)) \in P^{\mathcal{M}}$ (where $P \in \mathcal{L}$ is a k-ary predicate symbol and $t_i \in$ Term($\mathcal{L}$)).

- $\mathcal{M} \models_a t_1 \doteq t_2$ if and only if $\widetilde{a}(t_1) = \widetilde{a}(t_2)$ (where $t_1, t_2 \in$ Term($\mathcal{L}$)).

- $\mathcal{M} \models_a \neg\psi$ if and only if $\mathcal{M} \not\models_a \psi$.

- $\mathcal{M} \models_a (\psi \rightarrow \chi)$ if and only if $\mathcal{M} \not\models_a \psi$ or $\mathcal{M} \models_a \chi$.

- $\mathcal{M} \models_a \forall x_i \psi$ if and only if $\mathcal{M} \models_{a^*} \psi$ for all assignments a^* such that $a^*(x_j) = a(x_j)$ for all $j \neq i$.

Example 7.4.
Consider $\mathcal{M} = \langle \mathbb{Z}; \cdot \rangle$ as an $\{f\}$-structure (f a binary function symbol).
Let a be an assignment in $\mathbb{Z}$, and let

$$\phi = \forall x_0 \forall x_1 (f(x_0, x_2) \doteq f(x_1, x_2) \rightarrow x_0 \doteq x_1)$$

Then:

$$\begin{array}{ll} & \mathcal{M} \vDash_a \phi \\ \Leftrightarrow & \text{For all } a^* \text{ with } a^*(x_i) = a(x_i) \text{ for } i \neq 0, \\ & \mathcal{M} \vDash_{a^*} \forall x_1 (f(x_0, x_2) \doteq f(x_1, x_2) \rightarrow x_0 \doteq x_1). \\ \Leftrightarrow & \text{For all } a^{**} \text{ with } a^{**}(x_i) = a(x_i) \text{ for } i \neq 0, 1, \\ & \mathcal{M} \vDash_{a^{**}} (f(x_0, x_2) \doteq f(x_1, x_2) \rightarrow x_0 \doteq x_1). \\ \Leftrightarrow & \text{For all } a^{**} \text{ with } a^{**}(x_i) = a(x_i) \text{ for } i \neq 0, 1, \\ & \text{if } a^{**}(x_0) \cdot a^{**}(x_2) = a^{**}(x_1) \cdot a^{**}(x_2). \\ & \text{then } a^{**}(x_0) = a^{**}(x_1). \\ \Leftrightarrow & \text{For all } n, m \in \mathbb{Z}, \\ & \text{if } n \cdot a(x_2) = m \cdot a(x_2) \text{ then } n = m, \\ \Leftrightarrow & a(x_2) \neq 0. \end{array}$$

Notation 7.5. If a and a^* are assignments in a structure $\mathcal{M}$ and $i \in \mathbb{N}$, write $a^* \sim_i a$ to mean $a^*(x_j) = a(x_j)$ for all $j \neq i$.

So $\mathcal{M} \vDash_a \forall x_i \phi$ if and only if $\mathcal{M} \vDash_{a^*} \phi$ for all $a^* \sim_i a$.

Given $m \in M$, let $a[m/x_i]$ be the unique assignment such that $a[m/x_i] \sim_i a$ and $a[m/x_i](x_i) = m$, namely

$$a[m/x_i](x_j) = \begin{cases} a(x_j) & \text{if } j \neq i \\ m & \text{if } j = i. \end{cases}$$

Example 7.6.
Suppose $P \in \mathcal{L}$ with P a unary predicate symbol, $\mathcal{M}$ is an $\mathcal{L}$-structure,

$$\phi = (\forall x_0 P(x_0) \to P(x_1)),$$

and a is any assignment in $\mathcal{M}$. Then $\mathcal{M} \vDash_a \phi$.

Proof:
Suppose $\mathcal{M} \vDash_a \forall x_0 P(x_0)$.

Then for all $a^* \sim_0 a$, we have $\mathcal{M} \vDash_{a^*} P(x_0)$.
In particular, $\mathcal{M} \vDash_{a[a(x_1)/x_0]} P(x_0)$, so
$a(x_1) = a[a(x_1)/x_0](x_0) \in P^{\mathcal{M}}$.
So $\mathcal{M} \vDash_a P(x_1)$.

Hence $\mathcal{M} \vDash_a \phi$.

□

Definition 7.7.

Let $\mathcal{L}$ be a language.

- An $\mathcal{L}$-formula ϕ is **logically valid**, written $\models \phi$, if $\mathcal{M} \models_a \phi$ for *all* $\mathcal{L}$-structures $\mathcal{M}$ and for *all* assignments a in $\mathcal{M}$.

- $\phi \in \text{Form}(\mathcal{L})$ is **satisfiable** if $\mathcal{M} \models_a \phi$ for *some* $\mathcal{L}$-structure $\mathcal{M}$ and for *some* assignment a in $\mathcal{M}$.

- For $\Gamma \subseteq \text{Form}(\mathcal{L})$, we write $\mathcal{M} \models_a \Gamma$ to mean that $\mathcal{M} \models_a \phi$ for all $\phi \in \Gamma$.

- $\phi \in \text{Form}(\mathcal{L})$ is a **logical consequence** of $\Gamma \subseteq \text{Form}(\mathcal{L})$, written $\Gamma \models \phi$, if for *all* $\mathcal{L}$-structures $\mathcal{M}$ and for *all* assignments a in $\mathcal{M}$ with $\mathcal{M} \models_a \Gamma$, also $\mathcal{M} \models_a \phi$.

- $\phi, \psi \in \text{Form}(\mathcal{L})$ are **logically equivalent**, $\phi \models\!\!\!\!\dashv \psi$, if $\{\phi\} \models \psi$ and $\{\psi\} \models \phi$.

We abbreviate $\emptyset \models \phi$ to $\models \phi$; e.g. $\models (\forall x_0 P(x_0) \to P(x_1))$ by Example 7.6.

7.3. Some abbreviations

We use ...	as abbreviation for ...
$(\alpha \vee \beta)$	$((\alpha \to \beta) \to \beta)$
$(\alpha \wedge \beta)$	$\neg(\neg\alpha \vee \neg\beta)$
$(\alpha \leftrightarrow \beta)$	$((\alpha \to \beta) \wedge (\beta \to \alpha))$
$\exists x_i \phi$	$\neg\forall x_i \neg\phi$

Exercise 7.8.

For any $\mathcal{L}$-structure $\mathcal{M}$ and any assignment a in $\mathcal{M}$ one has

$$\begin{aligned}
\mathcal{M} \models_a (\alpha \vee \beta) &\Leftrightarrow \mathcal{M} \models_a \alpha \text{ or } \mathcal{M} \models_a \beta \\
\mathcal{M} \models_a (\alpha \wedge \beta) &\Leftrightarrow \mathcal{M} \models_a \alpha \text{ and } \mathcal{M} \models_a \beta \\
\mathcal{M} \models_a (\alpha \leftrightarrow \beta) &\Leftrightarrow \mathcal{M} \models_a \alpha \text{ iff } \mathcal{M} \models_a \beta \\
\mathcal{M} \models_a \exists x_i \phi &\Leftrightarrow \mathcal{M} \models_{a^*} \phi \text{ for some} \\
&\qquad \text{assignment } a^* \sim_i a
\end{aligned}$$

7.4. Tautologies

Let $\mathcal{L}$ be a first-order language.

Definition 7.9. A **tautology** of $\mathcal{L}$ is a substitution instance of a propositional tautology, i.e. a formula $\phi \in \text{Form}(\mathcal{L})$ obtained as follows:

- Let α be a logically valid formula of the <u>propositional</u> logic $\mathcal{L}_0$ with propositional variables among $p_0, \ldots, p_n$;
- let $\psi_0, \ldots, \psi_n \in \text{Form}(\mathcal{L})$;
- let ϕ be the $\mathcal{L}$-formula obtained from α by replacing each occurrence of p_i by ψ_i.

Example 7.10.
$(\forall x_0 P(x_0) \to (\neg x_0 \doteq x_1 \to \forall x_0 P(x_0)))$ is a tautology, obtained from the propositional validity $(p_0 \to (p_1 \to p_0))$.

Lemma 7.11. *Tautologies are logically valid: if ϕ is a tautology of $\mathcal{L}$, then $\models \phi$.*

Proof: Generally, let ϕ be the formula resulting from substituting ψ_i for p_i in a propositional formula α.
Given a structure $\mathcal{M}$ and assignment a, define a propositional valuation by

$$v(p_i) = T \;\Leftrightarrow\; \mathcal{M} \models_a \psi_i.$$

By the recursive definitions of $\models_a$ and of $\widetilde{v}$, it follows:

$$\widetilde{v}(\alpha) = T \;\Leftrightarrow\; \mathcal{M} \models_a \phi.$$

In particular, if α is a propositional validity, then $\widetilde{v}(\alpha) = T$ and $\mathcal{M} \models_a \phi$, so ϕ is logically valid.

□

Remark 7.12. Not all first-order logical validities are tautologies – e.g. $x_0 \doteq x_0$.

7.5. Free and bound variables

Recall from Example 7.4 that whether or not

$\langle \mathbb{Z}; \cdot \rangle \vDash_a \forall x_0 \forall x_1 (f(x_0, x_2) \doteq f(x_1, x_2) \to x_0 \doteq x_1)$

depends on $a(x_2)$. But it does not depend on $a(x_0)$ or $a(x_1)$.

This is because all occurrences of x_0 and x_1 in ϕ are subordinate to the corresponding quantifiers $\forall x_0$ and $\forall x_1$. We say that these occurrences are **bound**, while the occurrence of x_2 is **free**.

Definition 7.13.
Let $\mathcal{L}$ be a language, ϕ an $\mathcal{L}$-formula, and $x \in \{x_0, x_1, \ldots\}$ a variable.

An occurrence of x in ϕ is **free**, if

(i) ϕ is atomic; or

(ii) $\phi = \neg\psi$ resp. $\phi = (\chi \to \rho)$, and the occurrence of x is free in ψ resp. in χ or in ρ; or

(iii) $\phi = \forall x_i \psi$, and $x \neq x_i$, and the occurrence of x is free in ψ.

The variables which occur free in ϕ are called the **free variables of** ϕ,
Free(ϕ) := $\{x_i : x_i$ occurs free in $\phi\}$.

An occurrence which is not free is **bound**.
In particular, if $\phi = \forall x_i \psi$, then any occurrence of x_i in ϕ is bound. (We do not consider the use of the symbol x_i in the quantifier $\forall x_i$ as

an occurrence of x_i; e.g. x_0 does not occur in the formula $\forall x_0 c \doteq c$.)

Example 7.14.

$$(\exists x_0 P(\underbrace{x_0}_{bnd}, \underbrace{x_1}_{free}) \vee \forall x_1 (P(\underbrace{x_0}_{free}, \underbrace{x_1}_{bnd}) \rightarrow \exists x_0 P(\underbrace{x_0}_{bnd}, \underbrace{x_1}_{bnd})))$$

Lemma 7.15.
Let $\mathcal{L}$ be a language, let $\mathcal{M}$ be an $\mathcal{L}$-structure, let a_1 and a_2 be assignments in $\mathcal{M}$, and let ϕ be an $\mathcal{L}$-formula.

Suppose $a_1(x_i) = a_2(x_i)$ for every variable x_i with a free *occurrence in ϕ.*

Then

$$\mathcal{M} \vDash_{a_1} \phi \text{ iff } \mathcal{M} \vDash_{a_2} \phi.$$

Proof: For ϕ atomic: exercise.

Now use induction on the length of ϕ.
If $\phi = \neg\psi$ or $\phi = (\chi \to \rho)$, this is straightforward.

So say $\phi = \forall x_i \psi$, and assume the result holds for ψ.

Suppose $\mathcal{M} \vDash_{a_1} \forall x_i \psi$. We want to show $\mathcal{M} \vDash_{a_2} \forall x_i \psi$. So suppose $a_2^* \sim_i a_2$, and we want to show $\mathcal{M} \vDash_{a_2^*} \psi$.

Let $a_1^*(x_j) := a_1[a_2^*(x_i)/x_i]$. Then $\mathcal{M} \vDash_{a_1^*} \psi$, since $a_1^* \sim_i a_1$.

We conclude by applying the inductive hypothesis on ψ to obtain $\mathcal{M} \vDash_{a_2^*} \psi$ as required. For this, we need to show that if x_j occurs free in ψ then $a_2^*(x_j) = a_1^*(x_j)$.
If $j = i$, this is by definition of a_1^*.
If $j \neq i$, then x_j occurs free in ϕ, so

$$a_2^*(x_j) = a_2(x_j) = a_1(x_j) = a_1^*(x_j).$$

$\square$

Corollary 7.16.
Let $\mathcal{L}$ be a language, and let $\alpha, \beta \in \text{Form}(\mathcal{L})$. Assume the variable x_i has no free occurrence in α (i.e. $x_i \notin \text{Free}(\alpha)$). Then

$$\models (\forall x_i(\alpha \to \beta) \to (\alpha \to \forall x_i\beta)).$$

Proof:
Let $\mathcal{M}$ be an $\mathcal{L}$-structure and let a be an assignment in $\mathcal{M}$ such that $\mathcal{M} \models_a \forall x_i(\alpha \to \beta)$ and $\mathcal{M} \models_a \alpha$. We must show $\mathcal{M} \models_a \forall x_i\beta$.

So let $a^* \sim_i a$; we conclude by showing $\mathcal{M} \models_{a^*} \beta$.

Since $a^* \sim_i a$ and $\mathcal{M} \models_a \forall x_i(\alpha \to \beta)$, we have $\mathcal{M} \models_{a^*} (\alpha \to \beta)$.

But $\mathcal{M} \models_{a^*} \alpha$ by Lemma 7.15, since x_i is not free in α, so $\mathcal{M} \models_{a^*} \beta$ as required.

□

More generally, similar arguments yield the following.

Exercise 7.17. Assuming $x_i \notin \text{Free}(\alpha)$, the following logical equivalences hold:

- $(\alpha \vee \forall x_i \beta) \models\!\!\!\dashv \forall x_i(\alpha \vee \beta)$.
- $(\alpha \vee \exists x_i \beta) \models\!\!\!\dashv \exists x_i(\alpha \vee \beta)$.

7.6. Sentences

Definition 7.18.
An $\mathcal{L}$-formula σ with no free variables is called an $\mathcal{L}$-**sentence**. The set of all $\mathcal{L}$-sentences is denoted $\mathrm{Sent}(\mathcal{L})$.

By Lemma 7.15, for any $\mathcal{L}$-structure $\mathcal{M}$ and $\sigma \in \mathrm{Sent}(\mathcal{L})$, whether or not $\mathcal{M} \vDash_a \sigma$ does not depend on the choice of assignment a.

So we write

$$\mathcal{M} \vDash \sigma$$

if $\mathcal{M} \vDash_a \sigma$ for some (equivalently, all) a, and we then say that σ is **true** in $\mathcal{M}$, or $\mathcal{M}$ is a **model** of σ.

($\rightsquigarrow$ 'Model Theory')

Warning. The symbol '$\vDash$' is used in two quite distinct ways depending on what is on the left:

- Logical consequence: $\Gamma \vDash \phi$ where $\Gamma \subseteq \text{Form}(\mathcal{L})$;
- Satisfaction: $\mathcal{M} \vDash \sigma$, or $\mathcal{M} \vDash_a \phi$, where $\mathcal{M}$ is an $\mathcal{L}$-structure.

Many mathematical concepts can be naturally expressed by first-order formulas.

Example 7.19.

Let $\mathcal{L} = \{\cdot, e\}$ with $\cdot$ a binary function symbol and e a constant symbol.

Consider the sentences (writing x, y, z for x_0, x_1, x_2)

$$\begin{aligned}
\sigma_1 &: \ \forall x \forall y \forall z\ x \cdot (y \cdot z) \doteq (x \cdot y) \cdot x \\
\sigma_2 &: \ \forall x \exists y (x \cdot y \doteq e \land y \cdot x \doteq e) \\
\sigma_3 &: \ \forall x (x \cdot e \doteq x \land e \cdot x \doteq x)
\end{aligned}$$

Let $\mathcal{M} = \langle M; \cdot^{\mathcal{M}}, e^{\mathcal{M}} \rangle$ be an $\mathcal{L}$-structure. Then $\mathcal{M} \models \bigwedge_{i=1}^{3} \sigma_i$ if and only if $\mathcal{M}$ is a group.

Example 7.20.

Let $\mathcal{L} = \{E\}$ with E a binary relation symbol. Consider

$$\begin{aligned} \tau_1 &: \ \forall x E(x,x) \\ \tau_2 &: \ \forall x \forall y (E(x,y) \leftrightarrow E(y,x)) \\ \tau_3 &: \ \forall x \forall y \forall z (E(x,y) \to (E(y,z) \to E(x,z))) \end{aligned}$$

Then $\langle M; R \rangle \models \bigwedge_i \tau_i$ if and only if R is an equivalence relation on M.

Example 7.21.
Let $<$ be a binary predicate symbol, $\mathcal{L} := \{<\}$. Consider the sentence

$$\begin{aligned}\sigma_{\text{DLO}} := \forall x \forall y \forall z (\neg x < x \\ &\wedge (x < y \vee x \doteq y \vee y < x) \\ &\wedge ((x < y \wedge y < z) \rightarrow x < z) \\ &\wedge (x < y \rightarrow \exists w\, (x < w \wedge w < y)) \\ &\wedge \exists w\; w < x \\ &\wedge \exists w\; x < w).\end{aligned}$$

This axiomatises the **dense linear orders without endpoints**, i.e. they are precisely the models of σ.
In particular, $\langle \mathbb{Q}; < \rangle \vDash \sigma_{\text{DLO}}$ and $\langle \mathbb{R}; < \rangle \vDash \sigma_{\text{DLO}}$.

7.7. Isomorphism

Definition 7.22. Let $\mathcal{M} = \langle M; \ldots \rangle$ and $\mathcal{N} = \langle N; \ldots \rangle$ be $\mathcal{L}$-structures.

An **isomorphism** of $\mathcal{M}$ with $\mathcal{N}$ is a bijection $\theta : M \to N$ such that

- $\theta(c^{\mathcal{M}}) = c^{\mathcal{N}}$ for c a constant symbol;
- $\theta(f^{\mathcal{M}}(a_1, \ldots, a_k)) = f^{\mathcal{N}}(\theta(a_1), \ldots, \theta(a_k))$ for f a k-ary function symbol and $a_i \in M$;
- $(a_1, \ldots, a_k) \in P^{\mathcal{M}} \Leftrightarrow (\theta(a_1), \ldots, \theta(a_k)) \in P^{\mathcal{N}}$ for P a k-ary relation symbol and $a_i \in M$.

We write $\mathcal{M} \cong \mathcal{N}$ to mean that an isomorphism $M \to N$ exists.

Exercise 7.23. If $\mathcal{M} \cong \mathcal{N}$ and σ is an $\mathcal{L}$-sentence, then $\mathcal{M} \vDash \sigma$ if and only if $\mathcal{N} \vDash \sigma$.

7.8. Substitution

Let $\mathcal{M}$ be an $\mathcal{L}$-structure, $\phi \in \text{Form}(\mathcal{L})$, and suppose $\mathcal{M} \models \forall x_i \phi$.
If c is a constant symbol in $\mathcal{L}$, then $\mathcal{M} \models \phi[c/x_i]$ where $\phi[c/x_i]$ is the result of replacing each free instance of x_i in ϕ with c.

We would like to say more generally that

$$\models (\forall x_i \phi \to \phi[t/x_i])$$

for a term t, but we have to be careful:

Example 7.24.
Let $\mathcal{L}$ contain a constant symbol c, and let $\phi := \exists x_0 \neg x_0 \doteq x_1$.

Then $\mathcal{M} \models \forall x_1 \phi$ for any $\mathcal{L}$-structure $\mathcal{M}$ with at least two elements,
and then also $\mathcal{M} \models \phi[c/x_1] = \exists x_0 \neg x_0 \doteq c$.

However, if were to define $\phi[x_0/x_1]$ in the same way, we would obtain $\exists x_0 \neg x_0 \doteq x_0$, which does not hold in any $\mathcal{M}$.

Problem: the variable x_0 has become bound in the substitution.

Definition 7.25.
Given $\phi \in \text{Form}(\mathcal{L})$, a variable x_i, and a term $t \in \text{Term}(\mathcal{L})$, the result of **substituting t for x_i in ϕ** is the formula

$$(\phi)[t/x_i]$$

which is obtained by replacing each *free* occurrence of x_i in ϕ with the string t, *as long as* this does not lead to new bound occurrences of variables being introduced; if it does, we say that $(\phi)[t/x_i]$ is **undefined**.

For clarity, we restate this as a recursive definition:

- If ϕ is atomic, $(\phi)[t/x_i]$ is the result of replacing each instance of x_i in ϕ with t.
- $(\neg\psi)[t/x_i] := \neg(\psi)[t/x_i]$
  (undefined if $(\psi)[t/x_i]$ is).
- $((\psi \to \chi))[t/x_i] := ((\psi)[t/x_i] \to (\chi)[t/x_i])$
  (undefined if $(\psi)[t/x_i]$ or $(\chi)[t/x_i]$ is).
- $(\forall x_i \psi)[t/x_i] := \forall x_i \psi$.
- If $j \neq i$, $(\forall x_j \psi)[t/x_i] := \forall x_j (\psi)[t/x_i]$ unless x_j occurs in t and x_i occurs free in ψ, in which case $(\forall x_j \psi)[t/x_i]$ is undefined.

Notation 7.26. When no ambiguity could result, we often write $\phi[t/x_i]$ for $(\phi)[t/x_i]$.

Notation 7.27.
For a an assignment in an $\mathcal{L}$-structure and $t \in \text{Term}(\mathcal{L})$, let

$$a[t/x_i] := a[\widetilde{a}(t)/x_i].$$

Lemma 7.28 (Substitution Lemma)**.** *Let a be an assignment in an $\mathcal{L}$-structure $\mathcal{M}$. Let $\phi \in \text{Form}(\mathcal{L})$, $t \in \text{Term}(\mathcal{L})$, and suppose $\phi[t/x_i]$ is defined. Then*

$$\mathcal{M} \vDash_a \phi[t/x_i] \;\Leftrightarrow\; \mathcal{M} \vDash_{a[t/x_i]} \phi.$$

Proof: By induction on ϕ. So suppose the Lemma holds for shorter formulas ψ, i.e. $\mathcal{M} \vDash_a \psi[t/x_i] \Leftrightarrow \mathcal{M} \vDash_{a[t/x_i]} \psi$. We proceed by cases on the form of ϕ.

- ϕ atomic: Exercise.

- $\phi = \neg\psi$ or $\phi = (\chi \to \rho)$: Follows directly from IH.

- $\phi = \forall x_j \psi$: First, suppose $x_i \notin \text{Free}(\phi)$. Then $\phi[t/x_i] = \phi$, and a agrees with $a[t/x_i]$ on all $x \in \text{Free}(\phi)$, so we conclude by Lemma 7.15:

$$\mathcal{M} \vDash_{a[t/x_i]} \phi \Leftrightarrow \mathcal{M} \vDash_a \phi \Leftrightarrow \mathcal{M} \vDash_a \phi[t/x_i].$$

($\phi = \forall x_j \psi$, continued)

Otherwise, $x_i \in \text{Free}(\phi)$, and so $x_i \in \text{Free}(\psi)$ and $j \neq i$.
Since $\phi[t/x_i]$ is defined, x_j does not occur in t. Hence

$$\begin{aligned}\{a^*[t/x_i] : a^* \sim_j a\} &= \{a[m/x_j][t/x_i] : m \in M\} \\ &= \{a[t/x_i][m/x_j] : m \in M\} \\ &= \{a' : a' \sim_j a[t/x_i]\}, \qquad (*)\end{aligned}$$

where the second equality holds since x_j does not occur in t.

Now:

$$\begin{aligned}&\mathcal{M} \vDash_a \phi[t/x_i] \\ \Leftrightarrow\ &\mathcal{M} \vDash_a \forall x_j(\psi)[t/x_i] \\ \Leftrightarrow\ &\mathcal{M} \vDash_{a^*} \psi[t/x_i] \text{ for all } a^* \sim_j a \\ \Leftrightarrow\ &\mathcal{M} \vDash_{a^*[t/x_i]} \psi \text{ for all } a^* \sim_j a \text{ (by IH)} \\ \Leftrightarrow\ &\mathcal{M} \vDash_{a'} \psi \text{ for all } a' \sim_j a[t/x_i] \text{ (by } (*)) \\ \Leftrightarrow\ &\mathcal{M} \vDash_{a[t/x_i]} \phi.\end{aligned}$$

□

Corollary 7.29. *For any* $\phi \in \mathrm{Form}(\mathcal{L})$ *and* $t \in$ *Term*$(\mathcal{L})$ *such that* $\phi[t/x_i]$ *is defined,*

$$\vDash (\forall x_i \phi \to \phi[t/x_i]).$$

Proof: Let a be an assignment in an $\mathcal{L}$-structure $\mathcal{M}$.

Suppose $\mathcal{M} \vDash_a \forall x_i \phi$.
Then $\mathcal{M} \vDash_{a[t/x_i]} \phi$, since $a[t/x_i] \sim_i a$.
Hence $\mathcal{M} \vDash_a \phi[t/x_i]$ by the Substitution Lemma 7.28.

□

7.9. Prenex normal form

A formula is in **prenex normal form (PNF)** if it is of the form

$$Q_1 x_{i_1} Q_2 x_{i_2} \cdots Q_k x_{i_k} \phi',$$

where each Q_i is a quantifier (either $\forall$ or $\exists$), and ϕ' is a formula containing no quantifiers.

Theorem 7.30 (PNF Theorem)**.** *Every* $\phi \in \text{Form}(\mathcal{L})$ *is logically equivalent to an* $\mathcal{L}$*-formula in PNF.*

Proof: It suffices to prove this for ϕ which can be written using $\vee$ and $\neg$ as the only propositional connectives (rather than $\to$ and $\neg$), since by adequacy of $\mathcal{L}_{\text{prop}}[\neg, \vee]$, any $\mathcal{L}$-formula is logically equivalent to such a formula.

We prove this by induction on ϕ.

- ϕ atomic: ϕ is already in PNF.

- $\phi = \forall x_i \psi$. By inductive hypothesis, we may assume that ψ is in PNF. Then ϕ is already in PNF.

- $\phi = \neg\psi$. Again, we may assume that ψ is in PNF, say $\phi = \neg Q_1 x_{i_1} Q_2 x_{i_2} \cdots Q_k x_{i_k} \psi'$.

  Then by the equivalences
  $\neg\forall x_i \chi \vDash\Dashv \exists x_i \neg\chi$ and $\neg\exists x_i \chi \vDash\Dashv \forall x_i \neg\chi$,

  $$\phi \vDash\Dashv Q_1^- x_{i_1} \cdots Q_k^- x_{i_k} \neg\psi',$$

  where $Q_j^- := \begin{cases} \exists & \text{if } Q_j = \forall \\ \forall & \text{if } Q_j = \exists. \end{cases}$

- $\phi = (\psi \vee \chi)$. Again, we may assume ψ and χ are in PNF, say

$$\psi = Q_1 x_{i_1} \cdots Q_k x_{i_k} \psi'$$
$$\chi = Q'_1 x_{j_1} \cdots Q'_l x_{j_l} \chi'.$$

Note that $\forall x_i \alpha \vDash\!\dashv \forall x_j \alpha[x_j/x_i]$ if x_j does not appear in α.

Changing quantified variables in this way, we may assume that the variables quantified over in ψ (namely $x_{i_1}, \ldots, x_{i_k}$) do not appear in χ (quantified or not), and, similarly, the variables quantified over in χ (namely $x_{j_1}, \ldots, x_{j_l}$) do not appear in ψ.

But then by iterative application of Exercise 7.17 ("pulling the quantifiers out" of the disjunction),

$$\phi \vDash\!\dashv Q_1 x_{i_1} \cdots Q_k x_{i_k} Q'_1 x_{j_1} \cdots Q'_l x_{j_l} (\psi' \vee \chi').$$

□

8. Proofs

Associate to each first-order language $\mathcal{L}$ the formal system $S(\mathcal{L})$ with the following axioms and rules:

- **Axioms**: An **axiom** of $S(\mathcal{L})$ is an $\mathcal{L}$-formula of one of the following forms, where $\alpha, \beta, \gamma \in \text{Form}(\mathcal{L})$, $t \in \text{Term}(\mathcal{L})$, and $i, j \in \mathbb{N}$:

A1 $(\alpha \to (\beta \to \alpha))$

A2 $((\alpha \to (\beta \to \gamma)) \to ((\alpha \to \beta) \to (\alpha \to \gamma)))$

A3 $((\neg\alpha \to \beta) \to ((\neg\alpha \to \neg\beta) \to \alpha))$

A4 $(\forall x_i \alpha \to \alpha[t/x_i])$ where $\alpha[t/x_i]$ is defined

A5 $(\forall x_i(\alpha \to \beta) \to (\alpha \to \forall x_i \beta))$ where $x_i \notin \text{Free}(\alpha)$

A6 $x_i \doteq x_i$

A7 $(x_i \doteq x_j \to x_j \doteq x_i)$

A8 $(x_i \doteq x_j \to (\alpha \to \alpha[x_j/x_i]))$ where α is atomic

- **Rules**:

MP (Modus Ponens): From α and $(\alpha \to \beta)$ infer β.

Gen (Generalisation): For any variable x_k, from α infer $\forall x_k \alpha$.

In other words:

Definition 8.1.
Let $\Sigma \subseteq$ Sent($\mathcal{L}$). A formula $\phi \in$ Form($\mathcal{L}$) is **provable** in $S(\mathcal{L})$ from hypotheses Σ, written $\Sigma \vdash_{\mathcal{L}} \phi$ (or $\Sigma \vdash \phi$ for short), if there is a sequence of $\mathcal{L}$-formulas (a **derivation** or **proof**) $\phi_1, \ldots, \phi_n$ with $\phi_n = \phi$ such that for $i \leq n$, at least one of the following holds:

- (A1-A8) ϕ_i is an axiom of $S(\mathcal{L})$.
- (Hyp) $\phi_i \in \Sigma$.
- (MP) $\phi_k = (\phi_j \to \phi_i)$ for some $j, k < i$.
- (Gen) $\phi_i = \forall x_k \phi_j$ for some $j < i$ and some $k \in \mathbb{N}$.

Again, $\vdash \phi$ abbreviates $\emptyset \vdash \phi$.

Note: We define $\Sigma \vdash \phi$ only when Σ is a set of <u>sentences</u>.

Example 8.2 (Swapping variables)**.**
Let $\phi \in \text{Form}(\mathcal{L})$ be such that $\text{Free}(\phi) = \{x_i\}$ and $\phi[x_j/x_i]$ is defined. Then $\{\forall x_i \phi\} \vdash \forall x_j \phi[x_j/x_i]$.

Proof:

1	$\forall x_i \phi$	[Hyp]
2	$(\forall x_i \phi \to \phi[x_j/x_i])$	[A4]
3	$\phi[x_j/x_i]$	[MP 1,2]
4	$\forall x_j \phi[x_j/x_i]$	[Gen]

□

Remark 8.3. For any $\phi \in \text{Form}(\mathcal{L})$ and i, we have $\phi[x_i/x_i] = \phi$, and so $(\forall x_i \phi \to \phi)$ is an instance of A4.

Theorem 8.4 (Soundness Theorem). *For any* $\Sigma \subseteq \text{Sent}(\mathcal{L})$ *and* $\phi \in \text{Form}(\mathcal{L})$,

$$\Sigma \vdash \phi \ \Rightarrow \ \Sigma \vDash \phi.$$

Proof: First we show that A1-A8 are logically valid.

A1-3: These are tautologies, so are logically valid by Lemma 7.11.

A4: Corollary 7.29.

A5: Corollary 7.16.

A6-7: Immediate by reflexivity and symmetry of equality.

A8: Let $\mathcal{M}$ be an $\mathcal{L}$-structure and a an assignment in $\mathcal{M}$ such that

$$\mathcal{M} \vDash_a x_i \doteq x_j \text{ and } \mathcal{M} \vDash_a \alpha,$$

where α is atomic. We want to show that $\mathcal{M} \vDash_a \alpha[x_j/x_i]$.
Now $a(x_i) = a(x_j)$,
so $\widetilde{a}(t[x_j/x_i]) = \widetilde{a}(t)$ for any term t,
by induction on the length of t.

Now if $\alpha = P(t_1, \ldots, t_k)$, then

$$\begin{aligned}
\mathcal{M} \vDash_a \alpha &\Rightarrow (\widetilde{a}(t_1), \ldots, \widetilde{a}(t_k)) \in P^{\mathcal{M}} \\
&\Rightarrow (\widetilde{a}(t_1[x_j/x_i]), \ldots, \widetilde{a}(t_k[x_j/x_i])) \in P^{\mathcal{M}} \\
&\Rightarrow \mathcal{M} \vDash_a P(t_1[x_j/x_i], \ldots, t_k[x_j/x_i]) \\
&\Rightarrow \mathcal{M} \vDash_a \alpha[x_j/x_i],
\end{aligned}$$

as required. A similar argument applies if α is of the form $t_1 \doteq t_2$.

If $\sigma \in \Sigma$ is a hypothesis, then certainly $\Sigma \vDash \sigma$.

We can conclude by induction on the length of a proof, once we verify that the rules MP and Gen preserve the property of being a logical consequence of Σ.

MP: If $\Sigma \models \alpha$ and $\Sigma \models (\alpha \to \beta)$ then $\Sigma \models \beta$: indeed, for any $\mathcal{M}$ and a, if $\mathcal{M} \models_a \alpha$ and $\mathcal{M} \models_a (\alpha \to \beta)$ then $\mathcal{M} \models_a \beta$.

Gen: Suppose $\Sigma \models \psi$;
we want to show $\Sigma \models \forall x_i \psi$. Recall that the elements of Σ are sentences.

Let $\mathcal{M}$ be such that $\mathcal{M} \models \Sigma$,
and let a be an arbitrary assignment on $\mathcal{M}$.
We must show $\mathcal{M} \models_a \forall x_i \psi$.
So let $a^* \sim_i a$.
We must show $\mathcal{M} \models_{a^*} \psi$.
But since $\Sigma \models \psi$, we have $\mathcal{M} \models_{a'} \psi$ for *any* assignment a', in particular for a^*.

$\square$

Theorem 8.5 (Deduction Theorem). *Let* $\Sigma \subseteq \text{Sent}(\mathcal{L})$*, and* $\tau \in \text{Sent}(\mathcal{L})$*, and* $\phi \in \text{Form}(\mathcal{L})$*.*

$$\text{If } \Sigma \cup \{\tau\} \vdash \phi \text{ then } \Sigma \vdash (\tau \to \phi).$$

Proof: As for the deduction theorem of propositional logic, Theorem 4.6, we prove this by induction on the length of a proof. Axioms, hypotheses, and uses of MP are handled exactly as in that proof. To handle a use of Gen, deriving say $\forall x_i \chi$ from χ, it suffices to show:

Claim. *If* $\Sigma \vdash (\tau \to \chi)$ *then* $\Sigma \vdash (\tau \to \forall x_i \chi)$*.*

But indeed, $(\forall x_i(\tau \to \chi) \to (\tau \to \forall x_i \chi))$ is an instance of A5 since $x_i \notin \text{Free}(\tau) = \emptyset$, so the Claim follows by Gen and MP.

□

Lemma 8.6. *If ϕ is a tautology of $\mathcal{L}$, then $\vdash \phi$.*

Proof:
Say ϕ results from substituting ψ_i for p_i in the propositional validity α. By completeness of L_0, there is a proof $\alpha_1, ..., \alpha_{n-1}, \alpha$ of α in L_0.

Since A1, A2, A3 and MP are in $S(\mathcal{L})$, substituting ψ_i for p_i in each α_i yields a proof $\phi_1, ..., \phi_{n-1}, \phi$ in $S(\mathcal{L})$.

$\square$

By the lemma, we may freely introduce tautologies in our proofs in $S(\mathcal{L})$.

Example 8.7. Let $\tau \in \mathsf{Sent}(\mathcal{L})$ and $\psi \in \mathsf{Form}(\mathcal{L})$ with $\mathsf{Free}(\psi) \subseteq \{x_i\}$. Then

$$\vdash (\forall x_i(\psi \to \tau) \to (\exists x_i \psi \to \tau)).$$

Proof: Note that $\forall x_i(\psi \to \tau)$ is a sentence. We show

$$\{\forall x_i(\psi \to \tau)\} \vdash (\exists x_i \psi \to \tau);$$

the result then follows by the Deduction Theorem.

1	$\forall x_i(\psi \to \tau)$	[Hyp]
2	$(\forall x_i(\psi \to \tau) \to (\psi \to \tau))$	[A4]
3	$(\psi \to \tau)$	[MP 1, 2]
4	$((\psi \to \tau) \to (\neg\tau \to \neg\psi))$	[Tautology]
5	$(\neg\tau \to \neg\psi)$	[MP 3, 4]
6	$\forall x_i(\neg\tau \to \neg\psi)$	[Gen 5]
7	$(\forall x_i(\neg\tau \to \neg\psi) \to (\neg\tau \to \forall x_i \neg\psi))$	[A5]
8	$(\neg\tau \to \forall x_i \neg\psi)$	[MP 6, 7]
9	$((\neg\tau \to \forall x_i \neg\psi) \to (\neg\forall x_i \neg\psi \to \tau))$	[Tautology]
10	$(\neg\forall x_i \neg\psi \to \tau)$	[MP 8, 9]
11	$(\exists x_i \psi \to \tau)$	[Def. $\exists$]

In line 7, $x_i \notin \text{Free}(\neg\tau)$ because τ is a sentence, so the condition in A5 is met.

$\square$

9. Completeness and Compactness

Let $\mathcal{L}$ be a countable first-order language. Write $\vdash$ for $\vdash_{\mathcal{L}}$. We aim to show:

Theorem 9.1 (Gödel's Completeness Theorem)**.** *Let* $\Sigma \subseteq \text{Sent}(\mathcal{L})$ *and* $\phi \in \text{Form}(\mathcal{L})$*.*

$$\textit{If } \Sigma \models \phi \textit{ then } \Sigma \vdash \phi.$$

9.1. Proof of completeness

In outline, our proof strategy is much as in the propositional case:

- Reduce to: consistent $\Rightarrow$ satisfiable.
- Show: any consistent Σ extends to "complete witnessing" Σ'.
- Show: complete witnessing $\Rightarrow$ satisfiable.

This will be rather more involved than the propositional case. We begin with one easy reduction.

Remark 9.2. It suffices to prove Theorem 9.1 in the case that ϕ is a sentence.

Proof: Given $\Sigma \subseteq \text{Sent}(\mathcal{L})$ and $\phi \in \text{Form}(\mathcal{L})$ let $\text{Free}(\phi) = \{x_{i_1}, ..., x_{i_n}\}$ and set $\tau := \forall x_{i_1} ... \forall x_{i_n} \phi \in \text{Sent}(\mathcal{L})$.

Then if $\Sigma \models \phi$, then also $\Sigma \models \tau$, so $\Sigma \vdash \tau$ by Theorem 9.1 for sentences, but then $\Sigma \vdash \phi$ by A4 and MP (as in Remark 8.3).

$\square$

Definition 9.3. An $\mathcal{L}$-**theory** is a set $\Sigma \subseteq \text{Sent}(\mathcal{L})$ of $\mathcal{L}$-sentences.

Definition 9.4. Let $\Sigma \subseteq \text{Sent}(\mathcal{L})$ be an $\mathcal{L}$-theory.

- Σ is **consistent** (in $S(\mathcal{L})$) if for no $\chi \in \text{Sent}(\mathcal{L})$ do we have both $\Sigma \vdash \chi$ and $\Sigma \vdash \neg\chi$.

- Σ is **satisfiable** if it has a model, i.e. if there exists an $\mathcal{L}$-structure $\mathcal{M}$ with $\mathcal{M} \models \Sigma$.

Remark 9.5. If an $\mathcal{L}$-theory $\Sigma \subseteq \text{Sent}(\mathcal{L})$ is inconsistent, then $\Sigma \vdash \phi$ for *any* $\phi \in \text{Form}(\mathcal{L})$, since $(\chi \to (\neg\chi \to \phi))$ is a tautology.

Lemma 9.6. *Let Σ be an $\mathcal{L}$-theory and τ an $\mathcal{L}$-sentence.*

(i) $\Sigma \vdash \tau$ *if and only if* $\Sigma \cup \{\neg\tau\}$ *is inconsistent.*

(ii) $\Sigma \models \tau$ *if and only if* $\Sigma \cup \{\neg\tau\}$ *is unsatisfiable.*

Proof: Exactly as in Lemma 5.4, using the Deduction Theorem for (i).

$\square$

Given Lemma 9.6 and Remark 9.2, to prove Theorem 9.1 it suffices to prove:

Proposition 9.7. *Every consistent* $\mathcal{L}$*-theory is satisfiable.*

Note that the converse holds by soundness.

Definition 9.8.

- An $\mathcal{L}$-theory Σ is called **complete** (or *maximal consistent*) if Σ is consistent, and for any $\tau \in \text{Sent}(\mathcal{L})$: $\Sigma \vdash \tau$ or $\Sigma \vdash \neg\tau$.

- An $\mathcal{L}$-theory Σ is called **witnessing** if for all $\psi \in \text{Form}(\mathcal{L})$ such that $\Sigma \vdash \exists x_i \psi$ and $\exists x_i \psi \in \text{Sent}(\mathcal{L})$, there is some constant symbol $c \in \mathcal{L}$ such that $\Sigma \vdash \psi[c/x_i]$

Similar to the propositional case, we will prove Proposition 9.7 (consistent implies satisfiable) by first extending a consistent theory to a complete witnessing set, then showing that any complete witnessing set has a model. One slight complication will be that we have to add constants to the language to be the witnesses, but we will see that this is harmless.

Lemma 9.9. *Let Σ be a consistent $\mathcal{L}$-theory and τ an $\mathcal{L}$-sentence. Then either $\Sigma \cup \{\tau\}$ is consistent or $\Sigma \cup \{\neg\tau\}$ is consistent.*

Proof: Exactly as in Lemma 5.6.

$\square$

Lemma 9.10. *Let Σ be an $\mathcal{L}$-theory, ϕ an $\mathcal{L}$-formula, and $i \in \mathbb{N}$. Assume that $c \in \mathcal{L}$ is a constant symbol which does not occur in ϕ nor in any sentence in Σ, and that $\Sigma \vdash \phi[c/x_i]$.*

Then $\Sigma \vdash \phi$. Moreover, there is a proof of ϕ from Σ in which c does not appear.

Proof: Let $\alpha_1, \ldots, \alpha_n = \phi[c/x_i]$ be a proof of $\phi[c/x_i]$ from Σ. Let $m \in \mathbb{N}$ be such that that no α_k contains x_m.

Let $\alpha'_1, \ldots, \alpha'_n$ be the sequence obtained by replacing in each α_k each occurrence of c by x_m. We claim that $\alpha'_1, ..., \alpha'_n$ is a proof of $\alpha'_n = \phi[x_m/x_i]$ from Σ.

Since c occurs in no formula from Σ, if $\alpha_k \in \Sigma$ then $\alpha'_k = \alpha_k \in \Sigma$. If α_k is an axiom, then so is α'_k; this is immediate for all the axiom schemes except A4, and for A4 note that if $\alpha_k = (\forall x_j \phi \to \phi[t/x_j])$, then $\alpha'_k = (\forall x_j \phi' \to \phi'[t'/x_j])$ where ϕ' resp. t' is obtained by replacing c with x_m; here $\phi'[t'/x_j]$ is defined because $\phi[t/x_j]$ is and the quantifier $\forall x_m$ does not occur in ϕ'. The rules MP and Gen are insensitive to the change: if α_k follows from α_r and α_s by MP then α'_k follows from α'_r and α'_s by MP, and similarly with Gen.

So $\Sigma \vdash \alpha'_n = \phi[x_m/x_i]$. By Gen, this implies $\Sigma \vdash \forall x_m \phi[x_m/x_i]$.

Now note that $(\phi[x_m/x_i])[x_i/x_m]$ is defined and equal to ϕ: the effect of the substitutions is to replace the free occurrences of x_i in ϕ with x_m then revert them to x_i (which doesn't create new bound occurrences because only the free occurrences of x_i were substituted by x_m).

So by A4 with $t = x_i$ and MP, we obtain $\Sigma \vdash (\phi[x_m/x_i])[x_i/x_m] = \phi$, with a proof in which c does not appear.

□

Lemma 9.11. *Let Σ be a consistent $\mathcal{L}$-theory, and suppose $\Sigma \vdash \exists x_i \psi \in$ Sent($\mathcal{L}$), and c is a constant symbol of $\mathcal{L}$ which does* not *occur in ψ nor in any $\sigma \in \Sigma$.*
Then $\Sigma \cup \{\psi[c/x_i]\}$ is consistent.

Proof: We first show
Claim. *If τ is an $\mathcal{L}$-sentence in which c does not occur and $\Sigma \cup \{\psi[c/x_i]\} \vdash \tau$,*
then already $\Sigma \vdash \tau$.

So suppose $\Sigma \cup \{\psi[c/x_i]\} \vdash \tau \in$ Sent($\mathcal{L}$) and c does not occur in τ. Recall we also assumed that c does not occur in Σ or ψ.

Note that $\psi[c/x_i] \in$ Sent($\mathcal{L}$). By DT,
$\Sigma \vdash (\psi[c/x_i] \to \tau)$.
By Lemma 9.10, $\Sigma \vdash (\psi \to \tau)$.

By Gen, $\Sigma \vdash \forall x_i(\psi \to \tau)$.

Using Example 8.7, we obtain $\Sigma \vdash (\exists x_i \psi \to \tau)$.

But we assumed $\Sigma \vdash \exists x_i \psi$,
so by MP, $\Sigma \vdash \tau$, as required. This concludes the proof of the Claim.

Now if $\Sigma \cup \{\psi[c/x_i]\}$ were inconsistent then (by Remark 9.5) we would have for *any* τ that $\Sigma \cup \{\psi[c/x_i]\} \vdash \tau$ and $\Sigma \cup \{\psi[c/x_i]\} \vdash \neg\tau$; picking τ in which c does not occur, it would follow by the Claim that $\Sigma \vdash \tau$ and $\Sigma \vdash \neg\tau$, contradicting consistency of Σ.

$\square$

Lemma 9.12. *Let Σ be a consistent $\mathcal{L}$-theory, and suppose $\mathcal{L}$ contains infinitely many constant symbols not appearing in Σ. Then Σ extends to a complete witnessing $\mathcal{L}$-theory $\Sigma^* \subseteq \text{Sent}(\mathcal{L})$.*

Proof: Sent($\mathcal{L}$) is countable (by Fact 6.5); say $\text{Sent}(\mathcal{L}) = \{\tau_0, \tau_1, \tau_2, \ldots\}$.

We construct a chain $\Sigma_i \subseteq \text{Sent}(\mathcal{L})$

$$\Sigma = \Sigma_0 \subseteq \Sigma_1 \subseteq \Sigma_2 \subseteq \ldots$$

such that for each n:

(†) Σ_n is consistent, and $\mathcal{L}$ contains infinitely many constant symbols not appearing in Σ_n.

$\Sigma_0 := \Sigma$ satisfies (†) by assumption.

Given Σ_n satisfying (†), let

$$\Sigma_n' := \begin{cases} \Sigma_n \cup \{\tau_n\} & \text{if } \Sigma_n \cup \{\tau_n\} \text{ is consistent} \\ \Sigma_n \cup \{\neg\tau_n\} & \text{otherwise.} \end{cases}$$

Then Σ_n' is consistent by Lemma 9.9.

If $\tau_n \notin \Sigma_n'$ or if τ_n is not of the form $\exists x_i \psi$, let $\Sigma_{n+1} := \Sigma_n'$.

Otherwise, i.e. if $\tau_n = \exists x_i \psi \in \Sigma_n'$, choose a constant symbol $c \in \mathcal{L}$ which occurs in no formula in Σ_n' (such exists by (†)), and let $\Sigma_{n+1} := \Sigma_n' \cup \{\psi[c/x_i]\}$.
By Lemma 9.11, Σ_{n+1} is consistent.

Since $\Sigma_{n+1} \setminus \Sigma_n$ is finite, $\mathcal{L}$ contains infinitely many constant symbols not appearing in Σ_{n+1}. So Σ_{n+1} satisfies (†).

Finally, let $\Sigma^* := \bigcup_{n \geq 0} \Sigma_n$.

Then (as in Theorem 5.7) Σ^* is consistent since each Σ_n is,
and Σ^* is complete and witnessing by construction, since every sentence appears as some τ_n.

$\square$

Lemma 9.13. *Every complete witnessing $\mathcal{L}$-theory Σ is satisfiable.*

Moreover, Σ has a countable model (i.e. a model with countable domain).

Proof:
We prove this by a method known as the *Henkin construction*, named after its originator Leon Henkin.

A term is **closed** if no variable appears in it. Let T be the set of closed $\mathcal{L}$-terms. Define an equivalence relation E on T by

$$t_1 E t_2 \text{ iff } \Sigma \vdash t_1 \doteq t_2.$$

Let T/E be the set of equivalence classes t/E for $t \in T$. Note that T, and hence T/E, is countable.

Define an $\mathcal{L}$-structure $\mathcal{M}$ with domain T/E by

$$c^{\mathcal{M}} := c/E$$
$$f^{\mathcal{M}}(t_1/E, \ldots, t_k/E) := f(t_1, \ldots, t_k)/E$$
$$(t_1/E, ..., t_k/E) \in P^{\mathcal{M}} \Leftrightarrow \Sigma \vdash P(t_1, \ldots, t_k)$$

(for c a constant symbol, f a k-ary function symbol, and P a k-ary predicate symbol). We leave some verifications as exercises:

- E is indeed an equivalence relation on T. This follows from A6-8 and A4; see Sheet 4 Question 4(a).

- $f^{\mathcal{M}}$ and $P^{\mathcal{M}}$ are well-defined, i.e. if $t_i/E = t'_i/E$ for $i = 1, \ldots k$ then:

  - $f(t_1, \ldots, t_k)/E = f(t'_1, \ldots, t'_k)/E$,
  - $\Sigma \vdash P(t_1, \ldots, t_k) \Leftrightarrow \Sigma \vdash P(t'_1, \ldots, t'_k)$.

  This follows from A8 and A4; see Sheet 4 Question 4(b).

For $t \in T$, write $t^{\mathcal{M}}$ for $\widetilde{a}(t)$ where a is an arbitrary assignment (well-defined since t is closed). Then by a straightforward induction:

($\star$) $t^{\mathcal{M}} = t/E$ for any $t \in T$.

We conclude by showing $\mathcal{M} \vDash \Sigma$. In fact, we show more generally that for any $\tau \in \mathrm{Sent}(\mathcal{L})$,

$$\mathcal{M} \vDash \tau \;\Leftrightarrow\; \Sigma \vdash \tau.$$

We prove this by induction on the number of symbols among $\{\neg, \rightarrow, \forall\}$ in the sentence τ. We split into the possible cases for the form of τ:

- $\tau = P(t_1, ..., t_k)$. Then

$$\begin{aligned} \mathcal{M} \vDash \tau &\Leftrightarrow (t_1^{\mathcal{M}}, ..., t_k^{\mathcal{M}}) \in P^{\mathcal{M}} \\ &\Leftrightarrow (t_1/E, ..., t_k/E) \in P^{\mathcal{M}} \quad \text{(by } (\star)\text{)} \\ &\Leftrightarrow \Sigma \vdash \tau. \end{aligned}$$

- $\tau = t_1 \doteq t_2$. Then:

$$\begin{aligned} \mathcal{M} \vDash \tau &\Leftrightarrow t_1^{\mathcal{M}} = t_2^{\mathcal{M}} \\ &\Leftrightarrow t_1/E = t_2/E \quad \text{(by } (\star)\text{)} \\ &\Leftrightarrow t_1 E t_2 \\ &\Leftrightarrow \Sigma \vdash \tau. \end{aligned}$$

- $\tau = \neg\chi$:

$$
\begin{array}{lll}
 & \mathcal{M} \vDash \neg\chi & \\
\Leftrightarrow & \mathcal{M} \nvDash \chi & \text{[def. of `}\vDash\text{']} \\
\Leftrightarrow & \Sigma \nvdash \chi & \text{[IH]} \\
\Leftrightarrow & \Sigma \vdash \neg\chi & [\Sigma \text{ complete}]
\end{array}
$$

- $\tau = (\chi \to \rho)$:

$$
\begin{array}{lll}
 & \mathcal{M} \nvDash (\chi \to \rho) & \\
\Leftrightarrow & (\mathcal{M} \vDash \chi \text{ and } \mathcal{M} \nvDash \rho) & \text{[def. of `}\vDash\text{']} \\
\Leftrightarrow & (\Sigma \vdash \chi \text{ and } \Sigma \nvdash \rho) & \text{[IH]} \\
\Leftrightarrow & (\Sigma \vdash \chi \text{ and } \Sigma \vdash \neg\rho) & [\Sigma \text{ complete}] \\
\Leftrightarrow & \Sigma \vdash \neg(\chi \to \rho) & \text{[Tautology (see below)]} \\
\Leftrightarrow & \Sigma \nvdash (\chi \to \rho) & [\Sigma \text{ complete}]
\end{array}
$$

where the penultimate line uses the following tautologies:

$$(\chi \to (\neg\rho \to \neg(\chi \to \rho)))$$
$$(\neg(\chi \to \rho) \to \chi)$$
$$(\neg(\chi \to \rho) \to \neg\rho).$$

- $\tau = \forall x_i \phi$:
  By ($\star$), every element of the domain of $\mathcal{M}$ is the interpretation of some $t \in T$. So $\mathcal{M} \vDash \forall x_i \phi$ iff for all $t \in T$, $\mathcal{M} \vDash \phi[t/x_i]$; indeed:

$$\begin{aligned}
\mathcal{M} \vDash \forall x_i \phi &\Leftrightarrow \mathcal{M} \vDash_a \phi \text{ for all } a \\
&\Leftrightarrow \mathcal{M} \vDash_{a[t^{\mathcal{M}}/x_i]} \phi \text{ for all } a\text{, for all } t \in T \\
&\Leftrightarrow \mathcal{M} \vDash_a \phi[t/x_i] \text{ for all } a\text{, for all } t \in T \text{ (by} \\
&\Leftrightarrow \mathcal{M} \vDash \phi[t/x_i] \text{ for all } t \in T \text{ (since } \phi[t/x_i]
\end{aligned}$$

Now for $t \in T$, $\mathcal{M} \vDash \phi[t/x_i] \Leftrightarrow \Sigma \vdash \phi[t/x_i]$ by the IH, since $\phi[t/x_i] \in \text{Sent}(\mathcal{L})$ contains fewer symbols among $\{\neg, \rightarrow, \forall\}$ than $\tau = \forall x_i \phi$.

So to show $\Sigma \vdash \forall x_i \phi \Leftrightarrow \mathcal{M} \vDash \forall x_i \phi$, it suffices to show:

Claim. *$\Sigma \vdash \forall x_i \phi$ iff for all $t \in T$, $\Sigma \vdash \phi[t/x_i]$.*

$\Rightarrow$: A4 + MP.

$\Leftarrow$: First note:

$$\{\forall x_i \neg\neg\phi\} \vdash \forall x_i \phi; \qquad (\star)$$

indeed, by A4 we have $\{\forall x_i \neg\neg\phi\} \vdash \neg\neg\phi$; conclude using the tautology $(\neg\neg\phi \to \phi)$ and Gen.

Now suppose $\Sigma \nvdash \forall x_i \phi$.
Then $\Sigma \nvdash \forall x_i \neg\neg\phi$, by $(\star)$.
So by completeness, $\Sigma \vdash \neg\forall x_i \neg\neg\phi$,
i.e. $\Sigma \vdash \exists x_i \neg\phi$.
Note that $\exists x_i \neg\phi$ is a sentence, since $\tau = \forall x_i \phi$ is.
So since Σ is witnessing, $\Sigma \vdash \neg\phi[c/x_i]$ for some constant symbol c.
Then since Σ is consistent, $\Sigma \nvdash \phi[c/x_i]$.
But $c \in T$, so it is not the case that for all $t \in T$, $\Sigma \vdash \phi[t/x_i]$.

This concludes the proof of the Claim, and hence of the Lemma.

□

Finally, we deal with the problem that $\mathcal{L}$ might not have the additional constants required to build a witnessing set.

Let $C = \{c_0, c_1, ...\}$ be a countably infinite set of distinct symbols disjoint from $\mathcal{L}$, and define the extended language $\mathcal{L}' := \mathcal{L} \cup C$ in which each c_i is a constant symbol. Recall that we write $\Sigma \vdash_{\mathcal{L}'} \phi$ to mean that there exists a proof of ϕ from Σ in $S(\mathcal{L}')$ (meaning that the proof can use $\mathcal{L}'$-axioms). We continue to use $\vdash$ for provability in $S(\mathcal{L})$.

Lemma 9.14. *Let Σ be an $\mathcal{L}$-theory.*

(i) Suppose $\Sigma \vdash_{\mathcal{L}'} \tau \in \mathrm{Sent}(\mathcal{L})$. Then $\Sigma \vdash \tau$.

(ii) Suppose Σ is inconsistent in $S(\mathcal{L}')$. Then Σ is inconsistent in $S(\mathcal{L})$.

Proof:

(i) Since proofs are finite, for some n we have $\Sigma \vdash_{\mathcal{L}'_n} \tau$ where $\mathcal{L}'_n := \mathcal{L} \cup \{c_0, ..., c_{n-1}\}$.

If $n = 0$, we are done. Otherwise: $\tau = \tau[c_{n-1}/x_0]$ since τ is a sentence, so by Lemma 9.10, there is a proof in $S(\mathcal{L}'_n)$ of τ from Σ in which c_n does not appear, i.e. $\Sigma \vdash_{\mathcal{L}'_{n-1}} \tau$. So we conclude by induction on n.

(In other words: replacing the constants c_i in an $S(\mathcal{L}')$-proof of τ with distinct variables x_{j_i} not appearing in that proof yields a proof in $S(\mathcal{L})$.)

(ii) By Remark 9.5, for any $\tau \in \mathrm{Sent}(\mathcal{L})$ we have $\Sigma \vdash_{\mathcal{L}'} \tau$, and hence $\Sigma \vdash \tau$ by (i). So Σ is inconsistent in $S(\mathcal{L})$.

$\square$

Proof: [Proof of Proposition 9.7]

Let $\Sigma \subseteq \text{Sent}(\mathcal{L})$ be consistent in $S(\mathcal{L})$.

By Lemma 9.14(ii), Σ is also consistent in $S(\mathcal{L}')$.

By Lemma 9.12, Σ extends to a complete witnessing set $\Sigma^* \subseteq \text{Sent}(\mathcal{L}')$, which is satisfiable by Lemma 9.13. So say $\mathcal{M}'$ is an $\mathcal{L}'$ structure such that $\mathcal{M}' \models \Sigma^*$, so in particular $\mathcal{M}' \models \Sigma$.

Let $\mathcal{M}$ be the $\mathcal{L}$-structure obtained from $\mathcal{M}'$ by "forgetting" the new constants C.
Then $\mathcal{M} \models \Sigma$, as required.

$\square$

This concludes our proof of completeness.
Explicitly: *Proof:* [Proof of Theorem 9.1]

Given $\Sigma \subseteq \text{Sent}(\mathcal{L})$ and $\tau \in \text{Sent}(\mathcal{L})$,

$$\begin{array}{ll} & \Sigma \models \tau \\ \Rightarrow_{9.6(i)} & \Sigma \cup \{\neg\tau\} \text{ is unsatisfiable} \\ \Rightarrow_{9.7} & \Sigma \cup \{\neg\tau\} \text{ is inconsistent} \\ \Rightarrow_{9.6(ii)} & \Sigma \vdash \tau. \end{array}$$

It suffices to consider this case where τ is a sentence, by Remark 9.2.

□

To summarise, from soundness and completeness we conclude the following equivalences.

Proposition 9.15. *Let $\mathcal{L}$ be a countable first-order language, and let Σ be an $\mathcal{L}$-theory.*

(i) Σ is consistent if and only if it is satisfiable.

(ii) If ϕ is an $\mathcal{L}$-formula, then $\Sigma \vdash \phi$ if and only if $\Sigma \models \phi$.

9.2. Löwenheim-Skolem

By the moreover clause in Lemma 9.13, we obtain:

Theorem 9.16 (Weak downwards Löwenheim-Skolem Theorem)**.** *Every satisfiable set of sentences has a countable model.*

(The full Löwenheim-Skolem Theorem says somewhat more, and will be covered in C1.1 Model Theory.)

9.3. Compactness

We also deduce:

Theorem 9.17 (The Compactness Theorem for 1st-order logic)**.** *An $\mathcal{L}$-theory Σ is satisfiable if and only if every finite subset $\Sigma_0 \subseteq \Sigma$ is satisfiable.*

Proof: By Proposition 9.15(i) and finiteness of proofs (exactly as in the propositional case Theorem 5.11).

□

10. Applications

Throughout, $\mathcal{L}$ denotes a countable first-order language.

10.1. Axiomatisations

Definition 10.1. Let $\mathcal{M}$ be an $\mathcal{L}$-structure.

- The **(first-order) theory of** $\mathcal{M}$ is the $\mathcal{L}$-theory

  $$\mathsf{Th}(\mathcal{M}) = \mathsf{Th}^{\mathcal{L}}(\mathcal{M}) := \{\sigma \in \mathsf{Sent}(\mathcal{L}) \mid \mathcal{A} \vDash \sigma\},$$

  the set of all $\mathcal{L}$-sentences true in $\mathcal{M}$.

- An **axiomatisation** of $\mathsf{Th}(\mathcal{M})$ is a complete subset; i.e. a set of sentences which hold of $\mathcal{M}$ and which suffice to deduce any sentence which holds of $\mathcal{M}$.

Recall Hilbert's programme from Lecture 1. Now we have established the Completeness

Theorem, the programme would call for us to find "finitary" (i.e. computable) axiomatisations of the theories of mathematical structures.

In general this is *impossible*: Gödel's first incompleteness theorem shows that already the theory of arithmetic $\mathrm{Th}(\langle \mathbb{N}; +, \cdot \rangle)$ has no computable axiomatisation*. But for some interesting structures it is possible, as we will now begin to see.

10.2. Elementary equivalence

Definition 10.2. $\mathcal{L}$-structures $\mathcal{M}$ and $\mathcal{N}$ are **elementarily equivalent**, written $\mathcal{M} \equiv \mathcal{N}$, if $\mathrm{Th}(\mathcal{M}) = \mathrm{Th}(\mathcal{N})$.

*The precise mathematical definition of "computable" is outside the scope of this course, but intuitively, an axiomatisation Σ is computable if one can write a computer program which, given a (suitable encoding of a) sentence will tell us whether or not it is an element of Σ.

Elementary equivalence is weaker than isomorphism:

Remark 10.3. $\mathcal{M} \cong \mathcal{N}$ implies $\mathcal{M} \equiv \mathcal{N}$, by Exercise 7.23.

The converse fails; for example, if $\mathcal{M}$ is an uncountable $\mathcal{L}$-structure, then by Löwenheim-Skolem (Theorem 9.16) $\mathrm{Th}(\mathcal{M})$ has a countable model $\mathcal{N}$, and then $\mathcal{M} \equiv \mathcal{N}$ but $\mathcal{M} \ncong \mathcal{N}$.

Using the Completeness Theorem, we see:

Exercise 10.4. An $\mathcal{L}$-theory $\Sigma \subseteq \mathrm{Sent}(\mathcal{L})$ is complete if and only if Σ has a model and $\mathcal{M} \equiv \mathcal{N}$ for any two models $\mathcal{M}$ and $\mathcal{N}$ of Σ.

10.3. A criterion for completeness

Theorem 10.5 (Vaught's Test (countable case))**.** *Suppose* $\Sigma \subseteq \mathrm{Sent}(\mathcal{L})$ *has a unique countable model up to isomorphism, i.e.* Σ *is* *consistent and if $\mathcal{M}, \mathcal{N} \models \Sigma$ are countable then $\mathcal{M} \cong \mathcal{N}$.*

Then Σ is complete.

Proof. Let $\mathcal{M}, \mathcal{N} \models \Sigma$. We conclude by showing $\mathcal{M} \equiv \mathcal{N}$.

By Löwenheim-Skolem (Theorem 9.16), there are countable $\mathcal{M}' \equiv \mathcal{M}$ and $\mathcal{N}' \equiv \mathcal{N}$. Then $\mathcal{M}', \mathcal{N}' \models \Sigma$, so $\mathcal{M}' \cong \mathcal{N}'$, and so $\mathcal{M}' \equiv \mathcal{N}'$ by Remark 10.3. Hence $\mathcal{M} \equiv \mathcal{M}' \equiv \mathcal{N}' \equiv \mathcal{N}$. □

Remark 10.6. The converse fails. We will see an example in the next lecture.

Example 10.7. Let $\mathcal{L}_= := \emptyset$, the language with no non-logical symbols. For $n \geq 2$, set $\tau_n := \exists x_1 \ldots \exists x_n \bigwedge_{1 \leq i < j \leq n} \neg x_i \doteq x_j$. Then the models of

$$\Sigma_\infty := \{\tau_n : n \geq 2\}$$

are precisely the infinite $\mathcal{L}_=$-structures (i.e. the infinite sets). By Theorem 10.5, Σ_∞ is complete.

10.4. Example: Axiomatising $\mathrm{Th}(\langle\mathbb{Q};<\rangle)$

Recall from Example 7.21 the sentence σ_{DLO} axiomatising dense linear orders without endpoints.

Theorem 10.8 (Cantor). σ_{DLO} *has a unique countable model up to isomorphism (so any countable model is isomorphic to* $\langle\mathbb{Q};<\rangle$*).*

Proof. ("Back-and-forth argument")

Let $\mathcal{M},\mathcal{N} \vDash \sigma_{\mathrm{DLO}}$ be countable. By the non-existence of endpoints, each is infinite.

A **partial isomorphism** $\theta : \mathcal{M} \dashrightarrow \mathcal{N}$ is a partially defined injective map such that for all $a, b \in \text{dom}(\theta)$,

$$\mathcal{M} \vDash a < b \;\;\Leftrightarrow\;\; \mathcal{N} \vDash \theta(a) < \theta(b).$$

Enumerate the domains of $\mathcal{M}$ and $\mathcal{N}$ as $M = (m_i)_{i\in\mathbb{N}}$ and $N = (n_i)_{i\in\mathbb{N}}$ respectively. We recursively construct a chain of partial isomorphisms $\theta_i : \mathcal{M} \dashrightarrow \mathcal{N}$ such that

dom(θ_i) is finite, and for all $j < i$, $m_j \in \text{dom}(\theta_i)$ and n
(*)

Let $\theta_0 := \emptyset$, the empty map. This satisfies (*).

Given θ_i satisfying (*), we first extend θ_i by finding $n \in N$ such that setting $\theta_i'(m_i) := n$ yields a partial isomorphism $\theta_i' : \mathcal{M} \dashrightarrow \mathcal{N}$ with $\text{dom}(\theta_i') = \text{dom}(\theta) \cup \{m_i\}$.

Say $\mathrm{dom}(\theta_i) = \{a_1, \ldots, a_s\}$ with $\mathcal{M} \models a_k < a_l$ for $1 \leq k < l \leq s$. There are four cases:

(i) $m_i = a_k$ (some $k \in [1, s]$): set $n := \theta_i(a_k)$.

(ii) $m_i < a_1$: let $n \in N$ be such that $n < \theta_i(a_1)$ (n exists, since $\mathcal{N}$ has no endpoint).

(iii) $m_i > a_s$: let $n \in N$ be such that $n > \theta_i(a_s)$ (n exists, since $\mathcal{N}$ has no endpoint).

(iv) $a_j < m_i < a_{j+1}$ (some $j \in [1, s-1]$): let $n \in N$ be such that $\theta_i(a_j) < n < \theta_i(a_{j+1})$ (n exists, since $\mathcal{N}$ is dense).

In all cases, θ_i' is a partial isomorphism.

Symmetrically, $(\theta_i')^{-1} : \mathcal{N} \dashrightarrow \mathcal{M}$ extends to $\theta_i'' : \mathcal{N} \dashrightarrow \mathcal{M}$ with $n_i \in \mathrm{dom}(\theta_i)''$; then $\theta_{i+1} := (\theta_i'')^{-1} : \mathcal{M} \dashrightarrow \mathcal{N}$ is a partial isomorphism satisfying (*).

Then $\theta := \bigcup_i \theta_i : \mathcal{M} \xrightarrow{\cong} \mathcal{N}$ is a bijective partial isomorphism, i.e. an isomorphism. $\square$

Applying Theorem 10.5, we obtain:

Corollary 10.9. $\{\sigma_{\text{DLO}}\}$ *is complete. Hence* $\{\sigma_{\text{DLO}}\}$ *axiomatises* $\text{Th}(\langle\mathbb{Q};<\rangle)$.

Corollary 10.10. *Completeness of a linear order is not a first-order property: there is no* $\mathcal{L}_<$*-theory* Σ *such that the models of* Σ *are precisely the complete linear orders.*

Proof. Suppose such a Σ exists. Then $\langle\mathbb{R};<\rangle \vDash \Sigma$ since $\langle\mathbb{R};<\rangle$ is a complete linear order. But $\langle\mathbb{R};<\rangle \equiv \langle\mathbb{Q};<\rangle$, since both satisfy the complete theory $\{\sigma_{\text{DLO}}\}$, so then also $\langle\mathbb{Q};<\rangle \vDash \Sigma$. But $\langle\mathbb{Q};<\rangle$ is not a complete linear order, contradicting the desired property of Σ. $\square$

10.5. An algebraic application (non-examinable)

Let $\mathcal{L}_{\text{ring}} := \{+, -, \cdot, \bar{0}, \bar{1}\}$. Let ACF be the $\mathcal{L}_{\text{ring}}$-theory whose models are precisely the algebraically closed fields:

$$\text{ACF} := \text{ [Field axioms] } \cup \{\forall z_0 \ldots \forall z_n \left(\neg z_n \doteq \bar{0} \to \exists x \sum_{i=}^{n}\right.$$

Let

$$\text{ACF}_0 := \text{ACF} \cup \{\neg \bar{n} \doteq \bar{0} : n \geq 1\},$$

where for $n \geq 1$, $\bar{n}$ denotes the term $\bar{1} + \ldots + \bar{1}$ (n times). So the models of ACF_0 are precisely the algebraically closed fields of characteristic 0. In particular, $\langle \mathbb{C}; +, -, \cdot, 0, 1\rangle \vDash \text{ACF}_0$. We aim to show that ACF_0 is complete, i.e. axiomatises $\text{Th}(\langle \mathbb{C}; +, -, \cdot, 0, 1\rangle)$.

We can prove this analogously to the case of $\langle \mathbb{Q}; <\rangle$, but working with uncountable sets.

From now on, we assume the axiom of choice. We will explain this and the related notion of the **cardinality** ("size") $|A|$ of a set A in B1.2 Set Theory; for now it suffices to know that $|A| = |B|$ if and only if there exists a bijection $A \to B$, and cardinalities are linearly ordered.

Fact 10.11. *Any characteristic 0 algebraically closed field* $\langle K; +, -, \cdot, 0, 1\rangle \vDash \mathrm{ACF}_0$ *with the same cardinality as* $\mathbb{C}$ *is isomorphic to* $\langle \mathbb{C}; +, -, \cdot, 0, 1\rangle$.

Sketch proof. A subset A of a field is **algebraically independent** if there are no non-trivial polynomial relations between its elements, i.e. $f(a_1, \ldots, a_n) \neq 0$ for any $f \in \mathbb{Z}[X_1, \ldots, X_n] \setminus \{0\}$ and distinct elements $a_1, \ldots, a_n \in A$.

Then just as for linear independence in vector spaces: an algebraically closed field has a well-defined dimension ("transcendence degree") which is the cardinality of any maximal algebraically independent subset, this dimension determines an algebraically closed field of a given characteristic up to isomorphism, and the dimension of an uncountable ACF is equal to its cardinality. $\square$

Fact 10.12. *Let $\mathcal{L}'$ be a possibly uncountable first-order language, i.e. with sets of constant, function, and relation symbols of arbitrary cardinality. Let $|\mathcal{L}'|$ be the cardinality of the language, i.e. that of the alphabet.*

Let $\Sigma \subseteq \mathrm{Sent}(\mathcal{L}')$, and suppose any finite subset of Σ has a model. Then Σ has a model of cardinality (i.e. with domain of cardinality) $\leq |\mathcal{L}'|$.

Sketch proof. Our proof for countable languages mostly goes through directly. The only place we used the countability assumption was in extending a consistent set Σ to a complete witnessing set. This can be proven in the general case (assuming choice) as follows: by iteratively adding $|\mathcal{L}'|$ new constants, Σ extends to a consistent witnessing set; now the union of a chain of consistent witnessing sets containing Σ is still consistent and witnessing, so by Zorn's lemma there exists a maximal such with respect to inclusion, which is complete witnessing. □

Corollary 10.13. ACF_0 *is complete, hence axiomatises* $\mathrm{Th}(\mathbb{C})$*.*

Proof. Let $\mathcal{M} \vDash \mathrm{ACF}_0$. Note that $\mathcal{M}$ is infinite, since it has characteristic 0.

Let $C = \{c_a : a \in \mathbb{C}\}$ be a set of constant symbols of cardinality $|\mathbb{C}|$, and let

$\mathcal{L}' := \mathcal{L}_{\text{ring}} \cup C$. Let $\Sigma := \text{Th}^{\mathcal{L}_{\text{ring}}}(\mathcal{M}) \cup \{\neg c_a \doteq c_b : a, b \in \mathbb{C},\ a \neq b\} \subseteq \text{Sent}(\mathcal{L}')$. Then since $\mathcal{M}$ is infinite, any finite subset $\Sigma_0 \subseteq \Sigma$ is modelled by $\mathcal{M}$ with the finitely many c_a appearing in Σ_0 interpreted as distinct elements. So by Fact 10.12, Σ has a model $\mathcal{N}$ of cardinality $|\mathcal{N}| \leq |\mathcal{L}'| = |\mathbb{C}|$. Considering the interpretations of the c_a, we actually have $|\mathcal{N}| = |\mathbb{C}|$. Let $\mathcal{N}'$ be the $\mathcal{L}_{\text{ring}}$ structure obtained from $\mathcal{N}$ by ignoring the c_a. Then by Fact 10.11, $\mathcal{N}' \cong \mathbb{C}$. So $\mathcal{M} \equiv \mathcal{N}' \equiv \mathbb{C}$.

So we conclude that any two models of ACF_0 are elementarily equivalent, so ACF_0 is complete. □

Theorem 10.14 (Ax-Grothendieck)**.** *Let $F : \mathbb{C}^n \to \mathbb{C}^n$ be a polynomial map, i.e. $F(a_1, \ldots, a_n) = (F_1(a_1, \ldots, a_n), \ldots, F_n(a_1, \ldots, a_n))$, where $F_i \in \mathbb{C}[\overline{X}]$.*

If F is injective, then F is surjective.

Proof. Fact: The algebraic closure of the finite field $\mathbb{F}_p$ is the union of a chain of finite subfields, $\mathbb{F}_p^{\mathsf{alg}} = \bigcup_k \mathbb{F}_{p^{k!}}$.

Claim 10.15. *Let p be prime. Any injective polynomial map $F : (\mathbb{F}_p^{\mathsf{alg}})^n \to (\mathbb{F}_p^{\mathsf{alg}})^n$ is surjective.*

Proof. Let k_0 be such that the coefficients of F are in $\mathbb{F}_{p^{k_0!}}$.

Let $k \geq k_0$. Then $F((\mathbb{F}_{p^{k!}})^n) \subseteq (\mathbb{F}_{p^{k!}})^n$, and so by injectivity, finiteness of $(\mathbb{F}_{p^{k!}})^n$, and the pigeonhole principle, $F((\mathbb{F}_{p^{k!}})^n) = (\mathbb{F}_{p^{k!}})^n$.

Hence $F((\mathbb{F}_p^{\mathsf{alg}})^n) = (\mathbb{F}_p^{\mathsf{alg}})^n$. □

Let $d := \max_i \deg(F_i)$. Let σ be an $\mathcal{L}_{\mathsf{ring}}$-sentence expressing that any injective polynomial map $F : K^n \to K^n$ consisting of polynomials of degree $\leq d$ is surjective:

$$\sigma := \forall z_{1,0}, \ldots \forall z_{n,d} (\forall \overline{x} \forall \overline{y} ((\bigwedge_i \sum_j z_{i,j} x_i{}^j \doteq \sum_j z_{i,j} y_i{}^j) \to \bigwedge_i$$

$$\to \forall \overline{y} \exists \overline{x} \bigwedge_i \sum_j z_{i,j} x_i{}^j \doteq y_i).$$

Suppose $\mathbb{C} \not\models \sigma$. Then by completeness of ACF_0, $\mathsf{ACF}_0 \models \neg\sigma$. Then by compactness, for some $m \in \mathbb{N}$,

$$\mathsf{ACF} \cup \{\neg \bar{i} \doteq \bar{0} : 0 < i < m\} \models \neg\sigma.$$

So if $p > m$ is prime, $\mathbb{F}_p^{\mathsf{alg}} \models \neg\sigma$. But this contradicts the Claim. □